152309
NASA CR-152309
LSI TR-361-5

FAULT DIAGNOSTIC INSTRUMENTATION DESIGN FOR ENVIRONMENTAL CONTROL AND LIFE SUPPORT SYSTEMS

(NASA-CR-152309) FAULT DIAGNOSTIC INSTRUMENTATION DESIGN FOR ENVIRONMENTAL CONTROL AND LIFE SUPPORT SYSTEMS Final Report (Life Systems, Inc., Cleveland, Ohio.) 63 p HC A04/MF A01 CSCL 06K G3/54

N81-18656

Unclas
16503

FINAL REPORT

by

P. Y. Yang, K. C. You,
R. A. Wynveen and J. D. Powell, Jr.

October, 1979

Prepared Under Contract NAS2-10050

by

Life Systems, Inc.
Cleveland, OH 44122

for

AMES RESEARCH CENTER
National Aeronautics and Space Administration

TR-361-5

FAULT DIAGNOSTIC INSTRUMENTATION DESIGN FOR ENVIRONMENTAL CONTROL AND LIFE SUPPORT SYSTEMS

Final Report

by

P. Y. Yang, K. C. You,
R. A. Wynveen and J. D. Powell, Jr.

October, 1979

Prepared Under Contract NAS2-10050

by

LIFE SYSTEMS, INC.
Cleveland, Ohio 44122

for

AMES RESEARCH CENTER
NATIONAL AERONAUTICS AND SPACE ADMINISTRATION

FOREWORD

This report summarizes the development work of Advanced Instrumentation Fault Diagnostics for Regenerative Environmental Control and Life Support Systems conducted by Life Systems, Inc., during the period of October, 1978, to October, 1979, under NASA Contract NAS2-10050. The Program Manager was Dr. P. Y. Yang. Technical Support was provided by Dr. K. C. You, Dr. R. A. Wynveen, J. D. Powell, III, J. R. Gyorki, F. H. Schubert and J. D. Powell, Jr. Administrative and documentation support was provided by R. H. Kohler, B. A. Ginunas, D. A. Jones, B. M. Jaras and M. Prokopcak.

The authors wish to acknowledge the technical contributions, support and program guidance offered by the program Technical Monitor, P. D. Quattrone, Chief, Advanced Life Support Office, NASA, Ames Research Center.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ACRONYMS

A/D	Analog/Digital
ARS	Air Revitalization System
ARX-1	Air Revitalization System, One-Person Experimental
BIC	Built-in Checkout
BID	Built-in Diagnostic
C/M I	Control/Monitor Instrumentation
CPU	Central Processing Unit
CRT	Cathode-Ray Tube
DARS	Data Acquisition and Reduction System
EC/LSS	Environmental Control/Life Support System
EIA	Electronic Industries Association
EPROM	Erasable Programmable Read-Only Memory
I/O	Input/Output
LRC	Line Replaceable Component
LRU	Line Replaceable Unit
MDS	Microcomputer Development System
MTBF	Mean-Time-Between-Failures
MTTR	Mean-Time-To-Repair
R&D	Research and Development
RLSE	Regenerative Life Support Evaluation
TSA	Test Support Accessories

SUMMARY

Development of regenerative environmental control and life support systems requires instrumentation characteristics which evolve from one development phase to another. As the development phase moves toward flight hardware, the system availability becomes an important design aspect which requires high reliability and maintainability. As a part of the continuous development effort, a program to evaluate, design and demonstrate advanced instrumentation fault diagnostics was successfully completed at Life Systems, Inc. Fault tolerance designs for reliability and other instrumentation capabilities to increase maintainability were evaluated and studied.

The major accomplishments of the development program were:

- Fault tolerance design
- Built-in diagnostic design
- Actuator fault prediction evaluation and analysis
- Spare sensor calibration curve retention evaluation
- Automatic in situ sensor calibration study
- Data storing for fault isolation study
- Demonstration of intercomputer links with error checking capability
- Demonstration of microprocessor-based built-in diagnostic circuit

One of the most important benefits of the program is the elimination of the "instrumentation-lagging" syndrome -- the instrumentation is reaching its maturity at the same rate as the regenerative process development for the spacecraft atmosphere revitalization. The program also established commonality of instrumentation for environmental control and life support subsystems; in addition, it reduced the instrumentation development effort and the risks for individual subsystems as well as the integrated systems. The program provided a focal point for the instrumentation development effort and resulted in an extremely well coordinated program to ensure the instrumentation readiness for long-term manned space missions.

It was concluded from the studies and demonstrations conducted under this program that built-in circuits to test the integrity of electronic components are required to ensure system safety and ultimately to provide fault tolerance for higher reliability. It was also concluded that maintenance aids for better maintainability are required. Intercomputer communication with error checking and software recovery algorithms will enhance the instrumentation reliability. Redundancy at the component level is recommended because it provides higher reliability than that provided by redundancy at the system or assembly level. Further investigation of actuator signature analysis using audio signatures instead of vibration is strongly recommended. Spare sensor calibration curves should be retained at the sensor head as a part of the sensor using advanced electronic technology to minimize size, power consumption and maintenance effort. Automatic in situ sensor calibration is readily applicable to current, voltage, speed and combustible gas sensors. Use of a state transition approach for real-time diagnosis is recommended because of the smaller memory size requirement compared to the approach of recording and storing data for fault isolation diagnosis after a malfunction.

INTRODUCTION

Instrumentation is required to control and monitor Environmental Control/Life Support Systems (EC/LSS). The requirement of EC/LSS instrumentation begins with the essential functions to control the chemical, physical and electrochemical processes such as carbon dioxide (CO_2) removal, water electrolysis, CO_2 reduction and nitrogen (N_2) generation. As more sophisticated functions are introduced, safety functions to protect personnel and ensure equipment integrity are needed. Finally, the requirement for high reliability and maintainability and for low weight, size and power consumption will be imposed upon the designer by long duration flight missions.

Background

Regenerative EC/LSS processes have been under development for many years.(1-23) The EC/LSS consists of two major systems: the regenerative Air Revitalization System (ARS) and the Waste Water Management System (WWMS). Life Systems, Inc., (LSI) has been involved in the design, development and testing of ARS subsystems to remove excess moisture from the air, concentrate CO_2 from the air, reduce CO_2 to water and methane or carbon, generate oxygen (O_2) from water, resupply N_2 and provide N_2 and hydrogen (H_2) separation. In addition, LSI has also developed a separate Electrochemical Air Revitalization System (EARS) and a water reclamation subsystem using Vapor Compression Distillation (VCD) technology.

Because the applications of the National Aeronautics and Space Administration (NASA) requires low launch weight and long operating life based on in-flight maintenance, it is essential that the Control and Monitor Instrumentation (C/M I) development is in pace with the electrochemical and mechanical process development. This is especially important because the actual flight of the EC/LSS hardware is several years in the future. Because the technology associated with components of the electronic engineering field is expanding rapidly, the advancements projected for the electronics industry must be taken into consideration now when designing the C/M I for the advanced EC/LSS processes.

Previous Efforts

In general, instrumentation development efforts include the following eight areas:

1. Integration of subsystems into a complete system.
2. Development of instrumentation interior architecture including processor, logic, memory, input/output (I/O), signal conditioner, power conditioner, analog/digital (A/D) interface and power supply.
3. Development of Test Support Accessories (TSA) instrumentation and interfaces.

(1-23) References cited at the end of this report.

4. Development of operator/system interface.

5. Development of system maintenance aids.

6. Incorporation of advanced instrumentation concepts.

7. Incorporation of the developer's knowledge of operation.

8. Development of instrumentation packaging.

Figure 1 shows the relationship among these eight developmental areas. As shown in the figure, TSA development (not an instrumentation development area) is related to Areas 1 and 3. Subsystem component and performance evaluation is related to Area 7. Areas 1 through 5 have been addressed previously.[24-28] Areas 4 and 5, development of operator/system interface and system maintenance aids, received special attention under Contract NAS2-9251, "EC/LSS Maintenance Instrumentation."[25,26] The present Contract, NAS2-10050, addressed Area 6: incorporation of advanced instrumentation concepts.

Research and Development Type Control/Monitor Instrumentation

The EC/LSS instrumentation advanced through a series of subsystem development programs including CO_2 removal, CO_2 reduction, O_2 generation and N_2 generation.[6,11,14,15,17-22] The instrumentation trend is depicted in Figure 2. In the past, the C/M I was typically designed with hard-wired logic circuits with fault detection, built-in checkout and limited fault avoidance/prediction capability. Instrumentation adjustments were made through potentiometers and switches. Displays typically used multiple level indicators and panel meters.

The present generation of C/M I is designed around a minicomputer. Flexibility and operator/system interface are emphasized because it is primarily designed for the development and testing of EC/LSS process hardware under a laboratory research and development (R&D) environment. The present R&D type C/M I features cathode-ray tube (CRT) message display, advanced operator command keyboard, fault avoidance, fault prediction, fault detection, R&D flexibility and interface to a hard-copier/Data Acquisition and Reduction System (DARS). Based on this technology, a series of minicomputer-based, dedicated instrumentation hardware was developed for controlling and monitoring a variety of experimental EC/LSS hardware including an experimental, integrated one-person Air Revitalization System (ARX-1) and four subsystems for the three-person Regenerative Life Support Evaluation (RLSE) program. The RLSE subsystems involved were:

a. Electrochemical Depolarized CO_2 Concentrating Subsystem (CS-3)
b. Independent Air Revitalization Subsystem (IARS)
c. Vapor Compression Distillation Subsystem (VCDS)
d. Sabatier CO_2 Reduction Subsystem (S-CRS)

Figure 3 shows the ARX-1 C/M I and Figure 4 shows the four RLSE subsystem C/M I's. For communication and documentation purposes, these C/M I enclosures were designated Model 100 laboratory R&D C/M I.

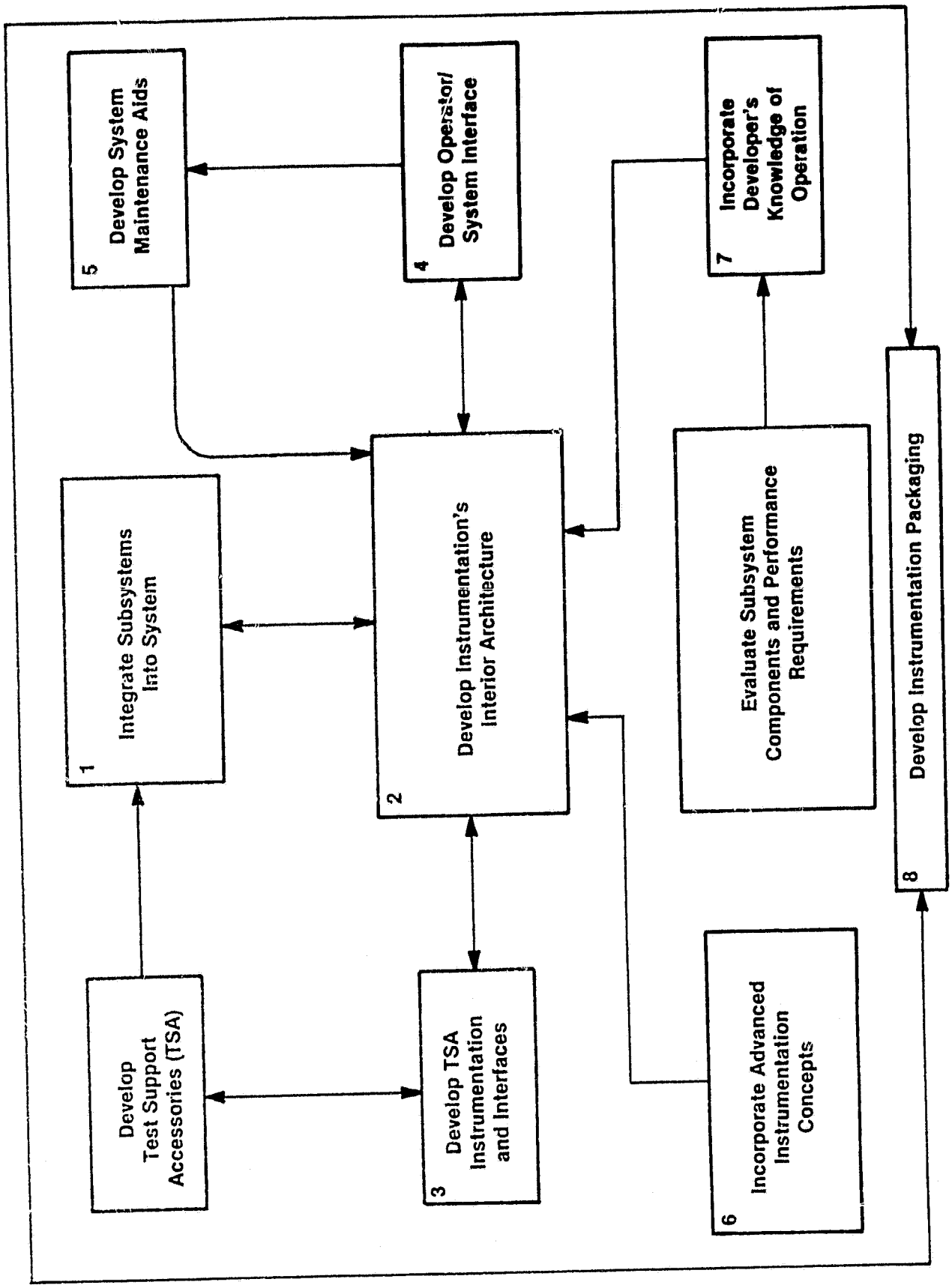
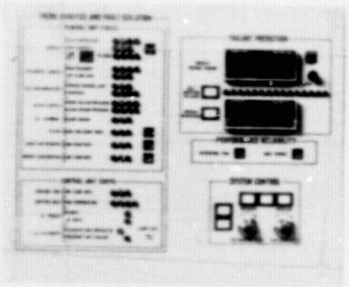
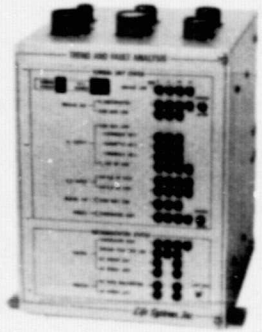


FIGURE 1 INSTRUMENTATION DEVELOPMENT AREAS

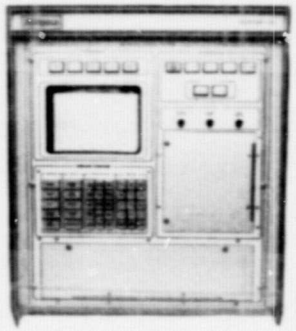
Past (1970-1976)

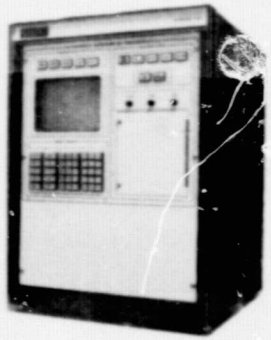




- Hardwired
- Four-level indicator
- Adjustment potentiometers/ switches
- Limited fault avoidance/ prediction
- Fault detection
- Built-in Checkout

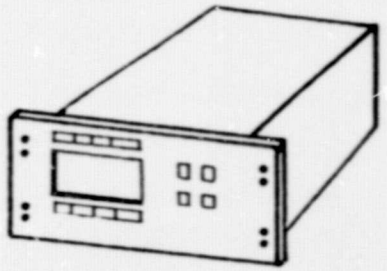
Present (1976-1980)

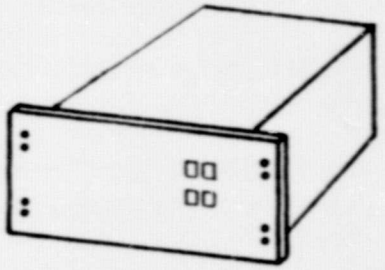




- Minicomputer-based
- CRT display
- Advanced operator command keyboard
- Fault avoidance/prediction
- Fault detection
- R&D flexibility
- Interface to hard-copier
- Interface to DARS

Future (Post 1980)





- Microprocessor-based
- Flight-oriented design
- Interface to central computer
- Fault avoidance/prediction
- Fault detection
- Maintenance aids
- Fault tolerance
- Built-in Diagnostic/ Checkout
- Advanced control techniques
- Smaller size
- Availability vs. weight penalty optimized

FIGURE 2 TREND OF EC/LSS INSTRUMENTATION

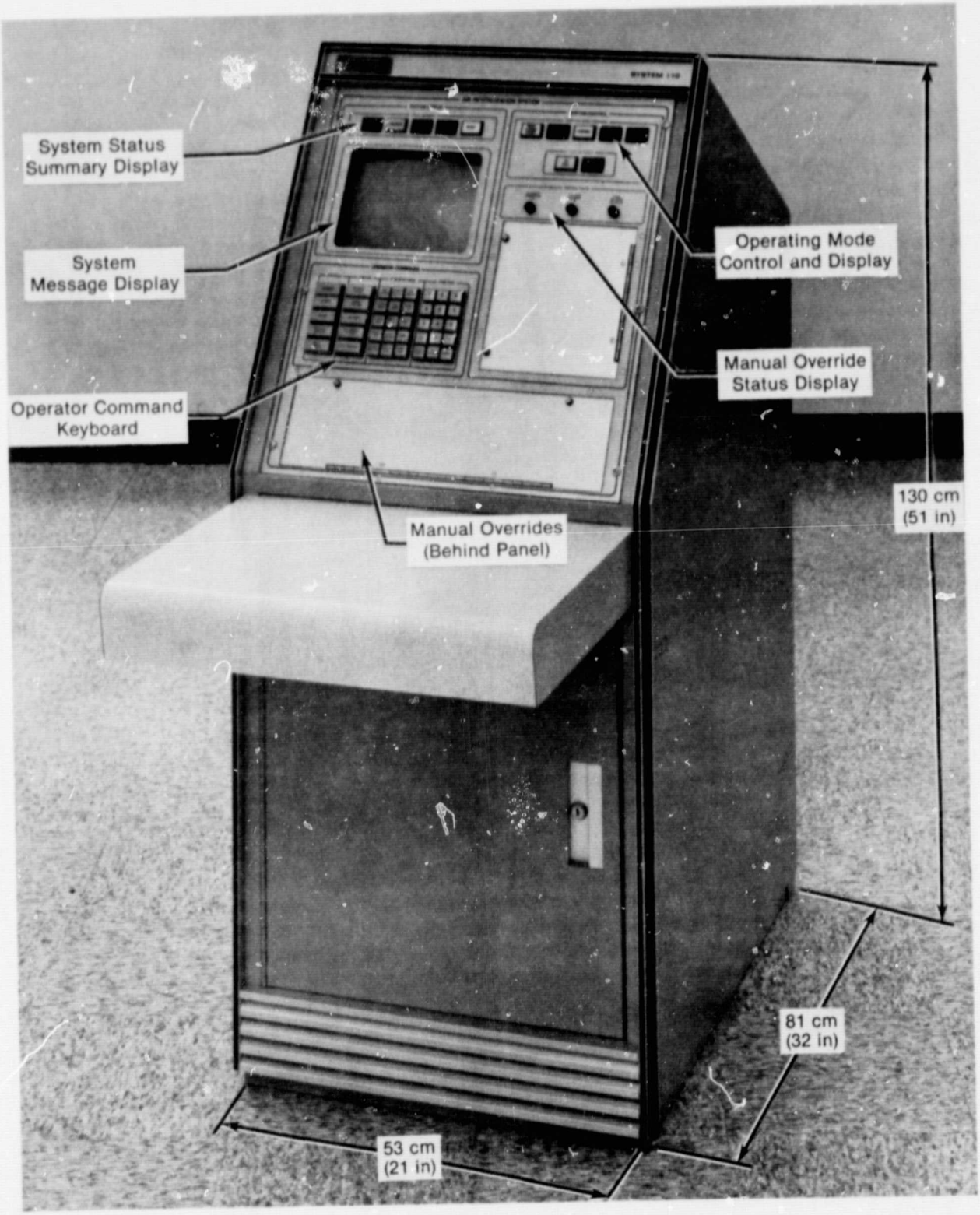


FIGURE 3 ARX-1 C/M I

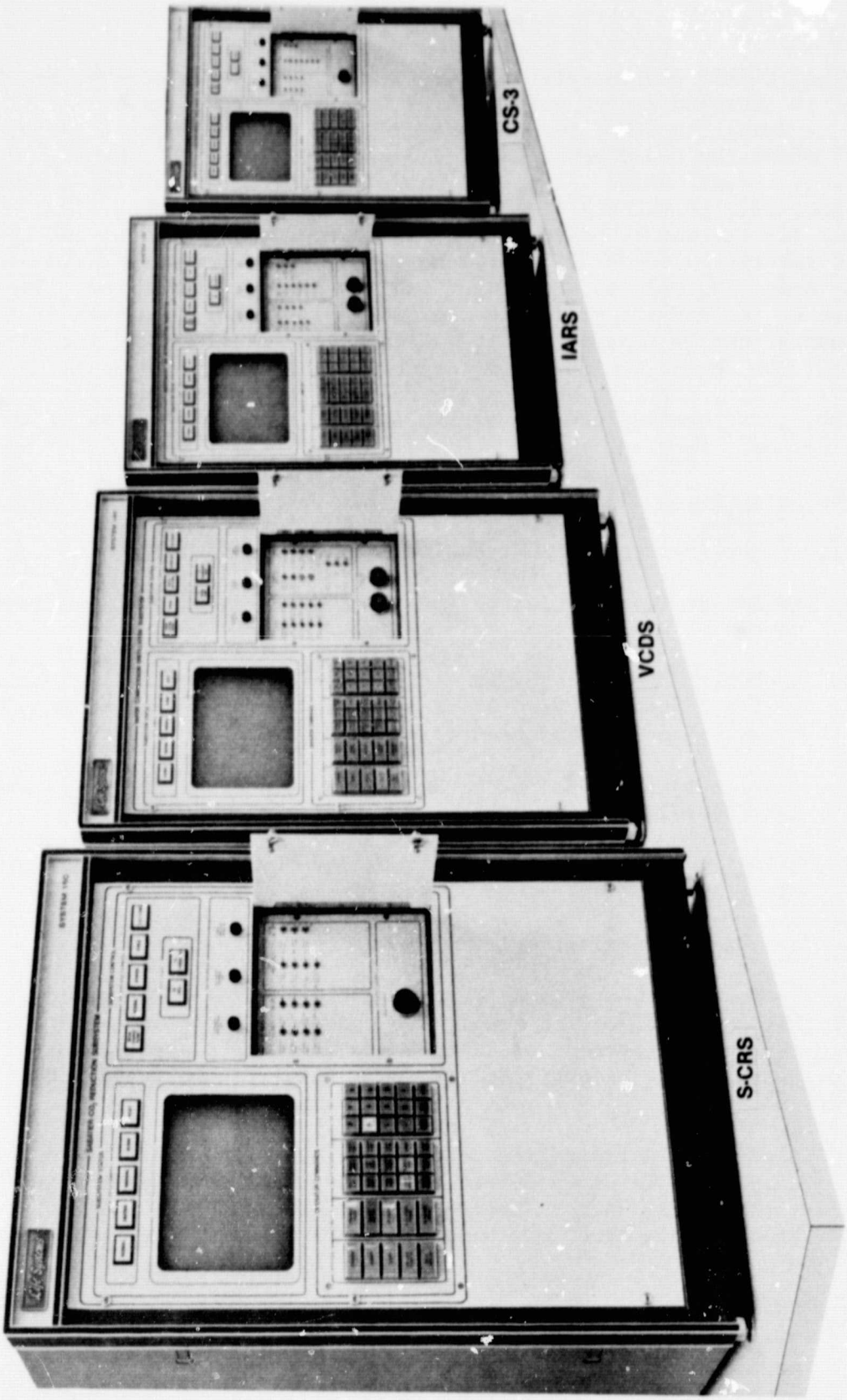


FIGURE 4 RLSE C/M I

Program Objectives

The overall EC/LSS instrumentation development program objectives are to reduce size and to increase system availability (reliability and maintainability). The goal is to ready the C/M I for the space flight mission several years from now. Figure 5 shows the two dimensions of the advanced C/M I R&D thrust. One is the development thrust toward the flight hardware C/M I. It is projected that this development will go through three generations with the present Model 100 being the first for laboratory breadboard EC/LSS hardware development and testing. The next generation, Model 200, will be dedicated to prototype hardware. Finally, Model 300 will be used for flight hardware applications. The other dimension of the R&D effort is the engineering thrust within each of the three C/M I generations to improve quality, eliminate weak links and increase capability but not change the instrumentation architecture. The space thrust is the major driving force to push development toward increasing capability per unit size, incorporating new components and concepts and increasing the availability per unit size.

Design Guidelines

The design guidelines established by the NASA included:

1. Employ design commonality for lower development cost and lower user cost.

2. Emphasize flexibility and development capability during the laboratory breadboard stages while allowing and requiring minimum effort to redesign for dedicated flight hardware.

3. Provide instrumentation hardware and techniques for users tnat do not have electronics or computer engineering background.

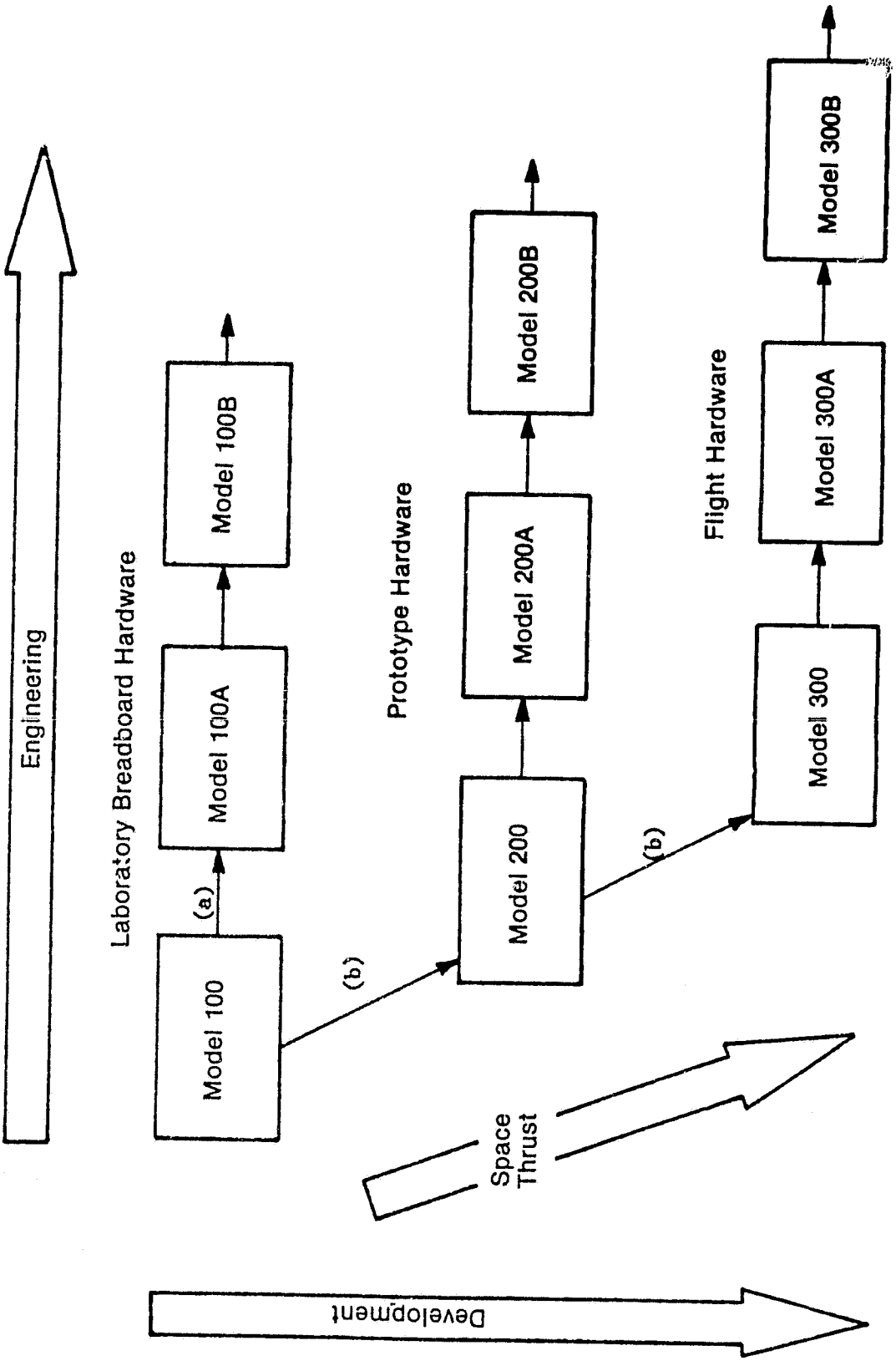
4. Allow expandability and compatibility for continuous upgrading as electronic technology advances.

Specific Tasks and Objectives

The specific tasks of the program were to evaluate and/or design advanced concepts in the fault diagnostics area. The concepts and design were those felt necessary for the advancement of instrumentation toward application of EC/LSS flight hardware with high availability and safety. These tasks and their objectives were to:

1. Study the incorporation of fault tolerance concepts which would enable the C/M I to detect and bypass faults within the instrumentation itself. A combination of techniques such as data/information transmission error checking and instrumentation redundancy were evaluated.

2. Evaluate the use of microprocessors to provide for checkout and diagnostic circuits to verify the integrity of the instrumentation and allow maintenance at the line replaceable component (LRC) or line replaceable unit (LRU) level.



(a) Improve quality, eliminate weak links and increase capability.
(b) Increase capability per unit size, reduce flexibility and incorporate new components and concepts. Increase availability per unit size.

FIGURE 5 ADVANCED EC/LSS C/M I DEVELOPMENT

3. Investigate the concept of using initial actuator signatures for periodic comparisons with the real-time actuator signatures as part of the fault prediction/isolation concept advancement.

4. Outline methods for and hardware size impacts of providing retention of calibration curves in the computer memory for spare sensors to be used following the isolation and replacement of a faulty sensor.

5. Specify the advantages of automatic in situ sensor calibration and recommend types of sensors applicable within an ARS.

6. Evaluate the need and techniques for recording operating parameters and conditions for each out-of-tolerance event occurring within the ARS hardware. Focus was on the time proceeding and following an out-of-tolerance event, allowing the data collected to be used for subsequent diagnostics.

7. Determine the most promising advanced C/M I approaches and demonstrate the capability.

FAULT DIAGNOSTICS DESIGN

Instrumentation for terrestrial, industrial processes typically limits its functions to measuring and controlling variables of the process or system as accurately as necessary.[29] These two functions, measurement and control, and the capability to interface with the operator or other instrument systems constitute the minimum and essential functions the instrumentation must perform.

In addition to the minimum functions, the C/M I must also provide monitor functions to ensure safety in case of any malfunctions. The safety aspects of the instrumentation include fault detection and fail-safe operation. More recently, fault isolation and sometimes fault correction instructions for the operator are included in industrial instrumentation because of the increasing demand for ease of maintenance.[30-35]

Additions to the essential and safety functions of C/M I are typically designed to increase the overall process reliability or maintainability to the level established by the applications specifications.[36-44] A high reliability means long duration operation without failures. A high maintainability means that if a failure does occur, it can be corrected in a short period of time. The combination of high reliability and high maintainability means high process availability. Instrumentation fault diagnostic functions can be incorporated to increase the process availability. In EC/LSS instrumentation design, the incorporation of fault diagnostic capabilities into the process hardware means adding more sensors, mechanical components and instrumentation to the flight hardware. An analysis, therefore, is needed to establish the fault diagnostic capabilities needed to increase process availability, the fault diagnostic design techniques available for EC/LSS application and the weight, volume and power consumption penalties associated with each additional capability.

Scope of Fault Diagnostics

Fault diagnostics typically include fault detection, fault isolation and fault correction instructions. The objectives of the fault diagnostics are to protect the system and personnel and to increase the system availability; therefore, a broader and preferred definition of fault diagnostics is "any functions designed to avoid, predict, detect, isolate or correct a component failure." Before a failure actually occurs the instrumentation should be designed to avoid as many faults as possible and to predict a failure when it has become unavoidable. The sequence of fault diagnostics thus begins with fault avoidance followed by fault prediction as shown in Table 1. When a failure has occurred, the fault detection function next in the sequence should convey to the operator that a failure has happened and then automatically trigger the maintenance aid functions: fault isolation and fault correction instructions. The ultimate goal of the instrumentation is to tolerate failures to a certain extent and maintain the system operation in spite of failures. The fault tolerance function is sometimes referred to as self-healing or self-correcting. A partial fault tolerance is called fail-soft or fail gracefully.

Reliability, Maintainability and Availability

To an EC/LSS design engineer, reliability is defined as "the probability that at a specified time, the EC/LSS is performing the functions as designed under specified conditions for an interval of duration."[45] Note that reliability is measured in terms of probability, expressed in meaningful quantitative terms and evaluated through applicable statistical methods. Being a probability, reliability is often stated as a positive number less than one, for example, 0.95. One of the classical reliability measures is mean-time-between-failures (MTBF). Time is an essential part of the reliability definition. For EC/LSS the term reliability often means the mission reliability which is the probability that the EC/LSS will operate without malfunctions for the duration of a mission.

For EC/LSS users, a more important system quality measurement is availability.[45] Availability is the probability that the system is operating satisfactorily at any point in time when used under stated conditions, where the total time considered includes operating time, active repair time, administrative time and logistic time. The mission availability (also called interval availability) is the expected fraction of a mission duration that the system will operate within the tolerances.

Availability is derived from reliability and maintainability.[45,46] Maintainability is the probability that when maintenance action is performed under stated conditions a failed system will be restored to operable conditions within a specified total downtime. A classic measure of maintainability is the mean-time-to-repair (MTTR).

Specific Tasks

The specific fault diagnostic areas addressed under this program were:

TABLE 1 SCOPE OF FAULT DIAGNOSTICS

Level	Function	Definition	Examples
1	Fault Avoidance	Prevent human errors in causing faults	Front panel human engineering, operator authorization codes, scheduled maintenance and automatic in situ calibration
2	Fault Prediction	Predict process or component failures by real-time analysis	Performance trend analysis, operator's knowledge incorporation, actuator signature analysis and real-time system diagnosis
3	Fault Detection	Detect symptoms of component failures not necessarily knowing the cause of the symptoms	Setpoint comparison, position/status indicators and built-in checkout/ diagnostic circuits
4	Fault Isolation	Triggered by fault detection to isolate causes of a symptom	Diagnosis software for failure analysis, actuator position/status indicators and operator's knowledge incorporation
5	Fault Correction Instructions	Instruct the operating personnel on the maintenance actions after a fault is detected	Display instructions "pressure too high, check valve V3," "check valve V1, if normal then check sensor P2," or component failure indicators
6	Fault Tolerance	Built-in capability to continue system operation without external assistance in the presence of failures	Triple redundant hydrogen sensors, adaptive control for nonoptimal environmental conditions and other redundant components

- Study fault tolerance concepts such as using transmission error checking and instrumentation redundancy
- Evaluate microprocessor-based self diagnostic circuit
- Investigate actuator fault prediction concept using signature analysis techniques
- Study spare sensor calibration curve retention
- Study automatic in situ sensor calibration
- Evaluate recording and storing of data for fault isolation
- Select and demonstrate the most promising one or two of the above approaches

The relationship between these tasks and the primary fault diagnostic levels is shown in Table 2. The potential impact of these functions on the instrumentation reliability, maintainability and availability are depicted in Table 3.

Fault Tolerance Concepts

Fault tolerance is the highest level fault diagnostic design which requires fault detection, isolation and self-repairing, adjustment or reconfiguration.

Transmission Error Checking. Transmission errors can be checked using information coding techniques. The following error checking techniques were evaluated for detecting transmission failures:

- Generate and compare a checkword of a number of data words
- Generate and compare odd or even parity of a word

The checkword can be generated by the summation of a block of data. In this case, the checkword is commonly referred to as the checksum of the data block. The checkword can also be generated by using other coding schemes such as cyclic redundancy code (CRC). In this case, the checkword is the coded dependent variable using a known equation with a block of data as the input independent variables to the equation.[47,48]

Parity checking is an established technique for digital data transmission error checking. In this technique, a parity bit is generated for each word transmitted to form an odd or even parity.

Transmission error checking alone is a fault detection function. Fault tolerance capability, however, can be achieved by issuing repetitive retrials after an error is detected. Retry of transmission is a form of software redundancy; and statistics have shown that 96% of peripheral storage to memory transmission errors can be corrected after three retries.[47]

For EC/LSS application, data and information are frequently transmitted back and forth between subsystems themselves as well as from subsystems to the central instrumentation. It is recommended that transmission error checking using the checksum technique be incorporated as a minimum. This capability is demonstrated as a part of this program effort and is discussed later in this report.

Use of Redundancies. A basic approach to fault tolerance is to use redundancies. As shown in Figure 6, dual redundancy provides fault detection but not tolerance.

TABLE 2 SPECIFIC TASKS VERSUS PRIMARY FAULT DIAGNOSTIC LEVELS

Tasks	Fault Diagnostic Levels[a]					
	FA	FP	FD	FI	FCI	FT
Fault Tolerance						√
Built-in Diagnostic			√			
Actuator Signature Analysis		√	√			
Calibration Curve Retention	√					
In Situ Calibration	√					
Data Storing for Fault Isolation				√		

(a) FA = Fault Avoidance, FP = Fault Prediction, FD = Fault Detection, FI = Fault Isolation, FCI = Fault Correction Instructions and FT = Fault Tolerance

TABLE 3 FUNCTIONS STUDIED AND THEIR PRIMARY RELATIONS TO SAFETY, RELIABILITY AND MAINTAINABILITY

Functions	Related to Safety	Related to Reliability	Related to Maintainability
Fault Tolerance		√	
Built-in Diagnostic	√	√	√
Actuator Signature Analysis			√
Calibration Curve Retention			√
In Situ Calibration		√	√
Data Storing for Fault Isolation			√

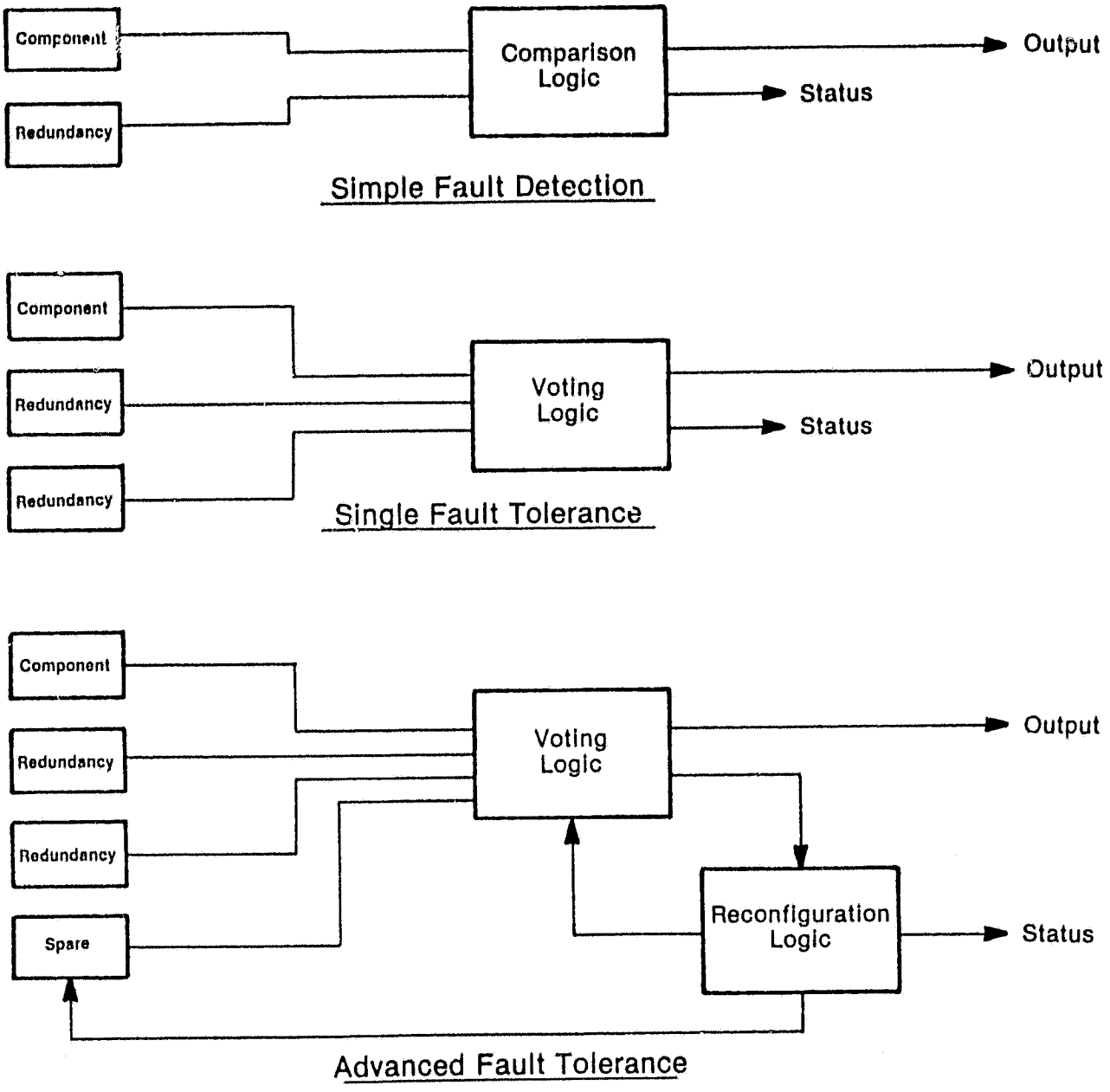


FIGURE 6 FAULT DETECTION AND TOLERANCE USING REDUNDANCY

Triple redundancy provides fault tolerance of a single failure. Triple redundancy with a spare provides fault tolerance of two failures.

Instead of redundancy comparisons, a signal conditioner built-in checkout (BIC) or a computer built-in diagnostic (BID) circuit can be used to detect failures. Using a BIC or BID circuit and a spare or redundancy, a fault tolerance of one failure can be achieved. Figure 7 depicts the configuration using BIC/BID.

Level of Redundancies. Redundancies can be implemented at different levels such as parts, components, assemblies and systems. Figure 8 shows the reliability curves of four different configurations resulting from a recent spaceborne fault tolerance study.(44) It was shown that using a single computer with voting logic and spares within the same computer will produce the highest reliability. This is illustrated in the calculations and comparison of Figure 9. Redundancies at assembly or system levels are sometimes required in order for the system recovery speed to a faulty component to be adequate. The total time between a failure and the completion of the recovery/reconfiguration sequence is critical for some applications such as spacecraft reentry or launch control computers. For EC/LSS, recovery speed is not critical and redundancies at the component level are feasible and recommended.

Built-in Diagnostic and Checkout Circuits

As shown in Figure 10, the concept of a BID is to use the computing and processing power of a microprocessor to detect failures of:

- Central Processing Unit (CPU)
- Memory
- Control and Data Bus
- A/D Interface
- Software

These five items are the key components of the EC/LSS C/M I. In addition, the BID can be designed to monitor critical process parameters directly.

A second built-in self-test function is performed by the BIC circuit. The concept of BIC is to perform active functional checkout testing of signal conditioners.

Both the BID and BIC circuits basically provide fault detection functions. Fault tolerance is achieved only when the BID/BIC are used with a redundant or spare component with reconfiguration logic circuits.

Fault isolation and maintenance aids can be provided by BID and BIC when additional circuits are incorporated to communicate with the operator and indicate the failed components. A typical application is shown in Figure 11. As a part of this program effort, a BID circuit was designed and demonstrated. This will be discussed later in this report.

Actuator Fault Prediction

A mechanical device usually produces sound and vibration when running. The characteristics of the sound or vibration reflect the uniqueness of the device

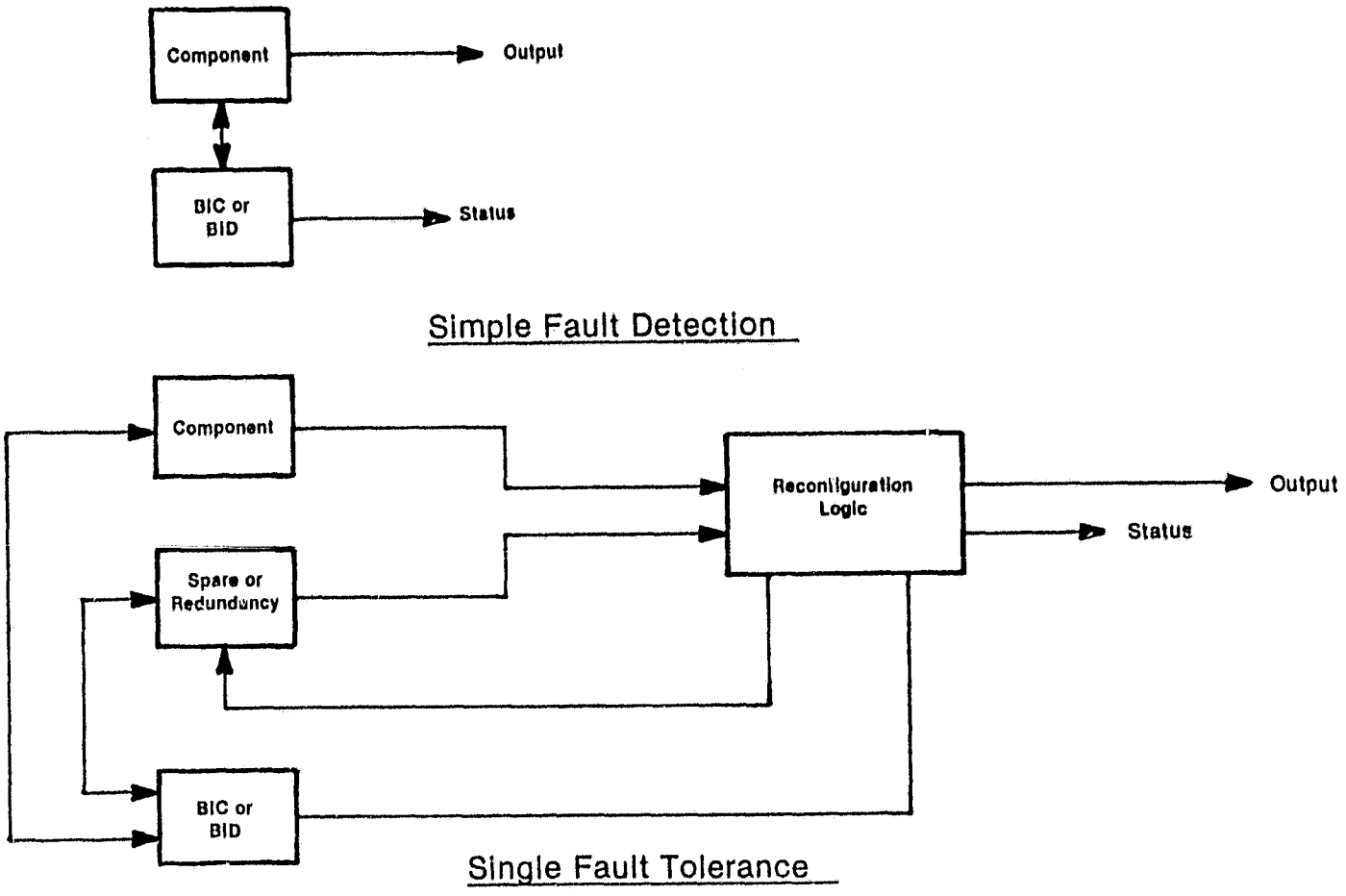
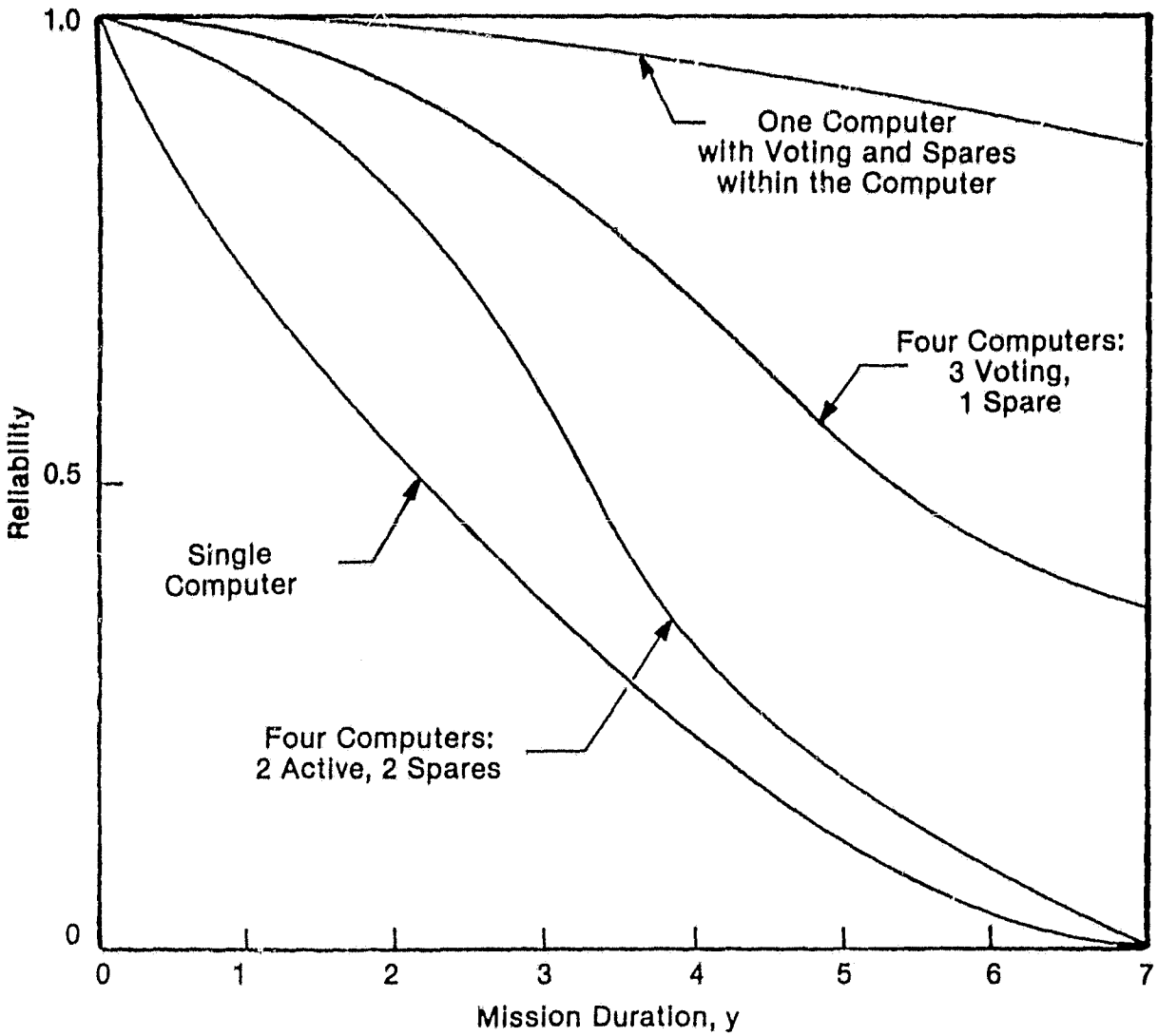


FIGURE 7 FAULT DETECTION AND TOLERANCE USING BIC/BID



Source: Space-Borne Fault Tolerance Study by Rathcon[44]

FIGURE 8 RELIABILITY CURVES

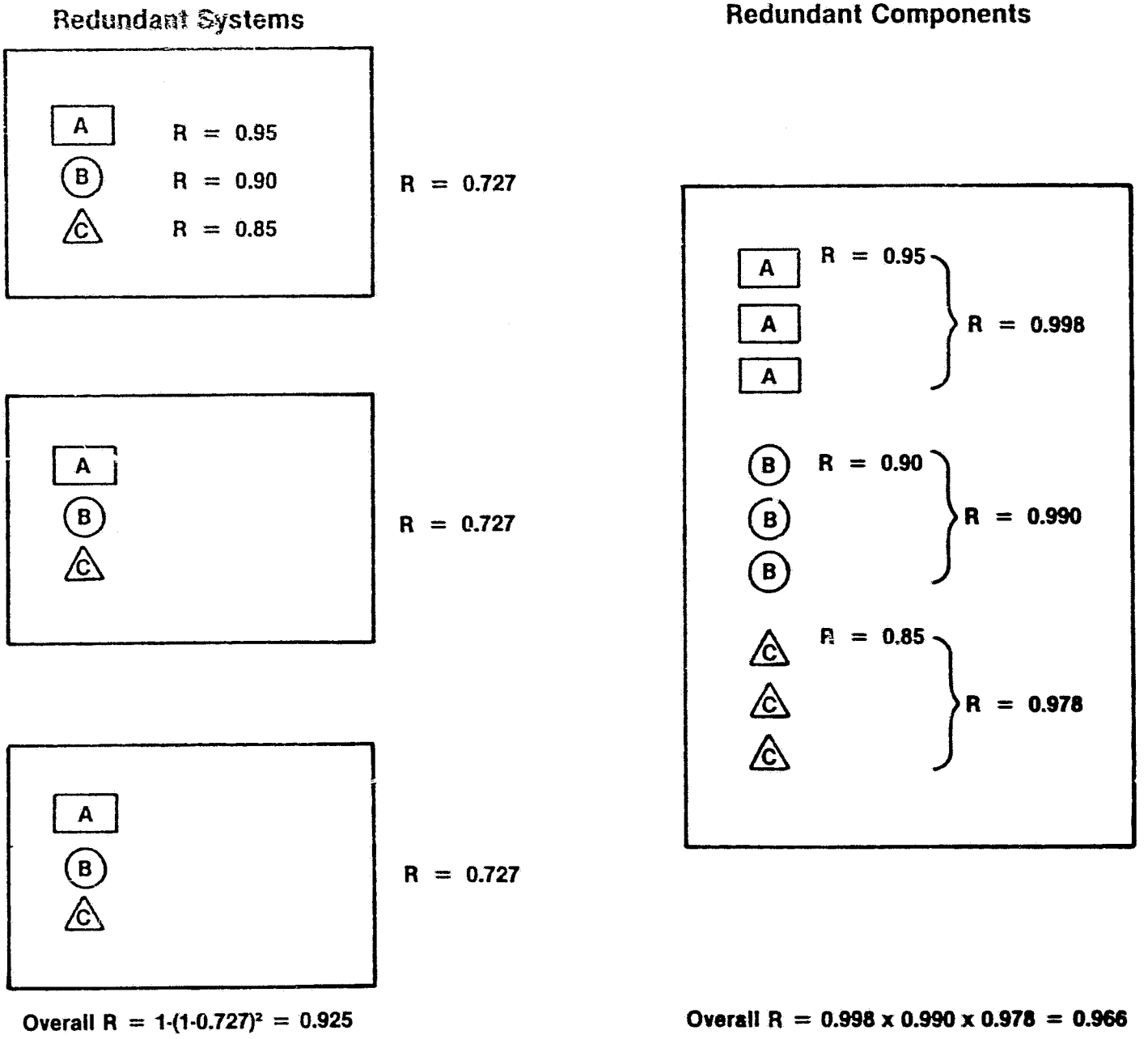


FIGURE 9 RELIABILITY COMPARISON EXAMPLE

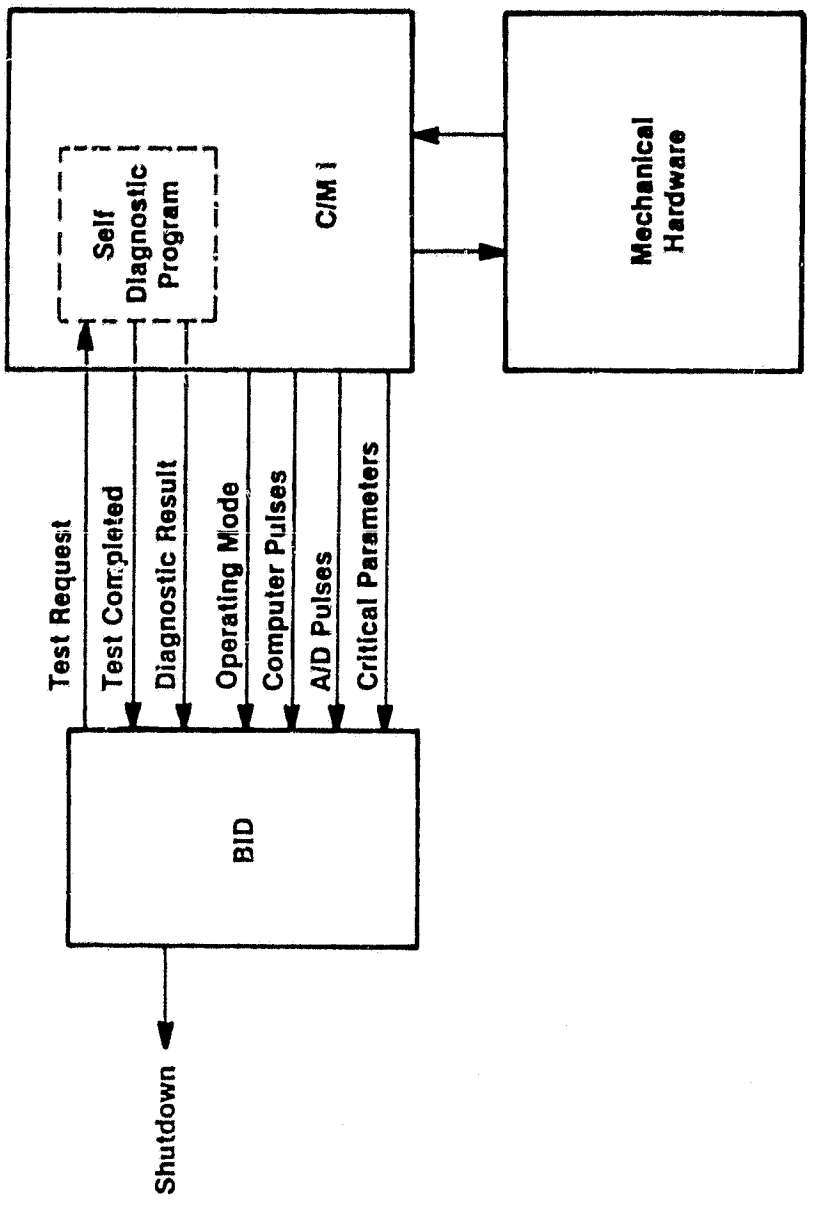


FIGURE 10 MICROPROCESSOR-BASED BID CIRCUIT

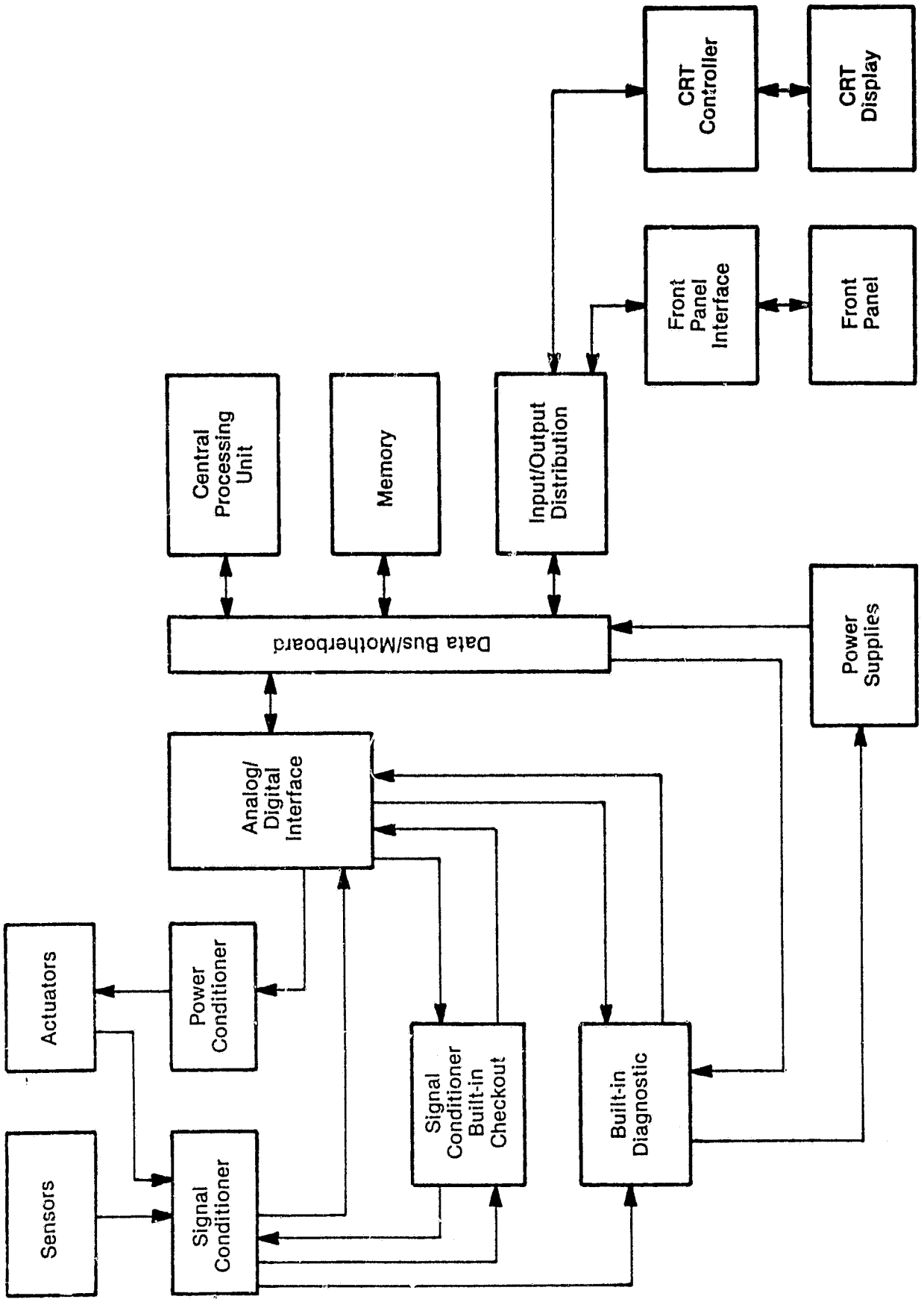


FIGURE 11 C/M I WITH BID AND CHECKOUT CIRCUITS

and its running condition. In other words, the sound or vibration produced by a machine contains information that can be used to judge the condition of the machine. There are some commercially available vibration monitors using velocity and accelerometer type vibration transducers. Monitoring the vibration pattern requires a number of transducers installed at different places in the machine to detect vibrations in different directions and of different parts. Most vibration sensors are invasive and have to be attached to internal parts of the machine. Sound monitoring has the advantage of not requiring many sensors distributed in the system and not invasive; but it has the disadvantage of being more susceptible to background noises. An audio signature analyzer which utilizes pattern recognition techniques[49] to detect or predict the fault of a running machine by its audio signature will have a better maintainability and, because of the fewer transducers involved, a better hardware reliability.

Figure 12 explains the concept of the analyzer. Microphones or audio sensors are used to obtain the audio waveform generated by the running device such as a motor. The audio information passing through the signal conditioning stage and digitization stage is converted into digital signals and then sent into a computer for processing. The pattern recognition technique includes two steps. In the first step, the computer extracts the information from an input pattern. The extracted information expressed as numbers, usually called features, should contain the unique characteristics of the input pattern. In the second step, the computer analyzes the pattern features and outputs its judgment. Figure 13 presents the block diagram of the process.

Audio Pattern Recognition. An audio waveform generated by a running machine is a time-varying function. At any time instant, the audio pattern can be analyzed as a function of frequency (an audio spectrum). Figure 14 illustrates an audio pattern in three-dimensional space. For simplicity of discussion, the audio spectrum is assumed to be stationary, that is, the audio spectrum is independent of time.

For a computer to process the signal, the continuous audio spectrum A_t, see Figure 14, should be quantized along the frequency axis. Then, the magnitude at each frequency should be converted into a digital number. The pattern can now be described as a discrete pattern D_t which is a vector of k variables.

$$D_t = \left[M_{tf1}, M_{tf2}, \ldots\ldots\ldots\ldots M_{tfk} \right] \qquad (1)$$

Because of random noise, the distribution of the discrete patterns of normal conditions usually forms a k-dimensional Gaussian function. Figure 15 shows the probability density function of one-dimensional Gaussian distribution. The probability density function of k-dimensional Gaussian distribution[50] is as follows:

$$P_k(D_t) = \frac{1}{(2^k \pi^k)^{\frac{1}{2}} |A^{-1}|^{\frac{1}{2}}} \exp -\tfrac{1}{2}(D_t - D_o) A (D_t - D_o)^T \qquad (2)$$

A = Covariance Matrix
D_o = Pattern Mean = $\left[M_{f1}, M_{f2} \ldots\ldots\ldots M_{fk} \right]$
T = Transpose of Matrix

Figure 16 shows the probability density functions of normal patterns and abnormal patterns. If the pattern occurs in the normal region, it implies that the

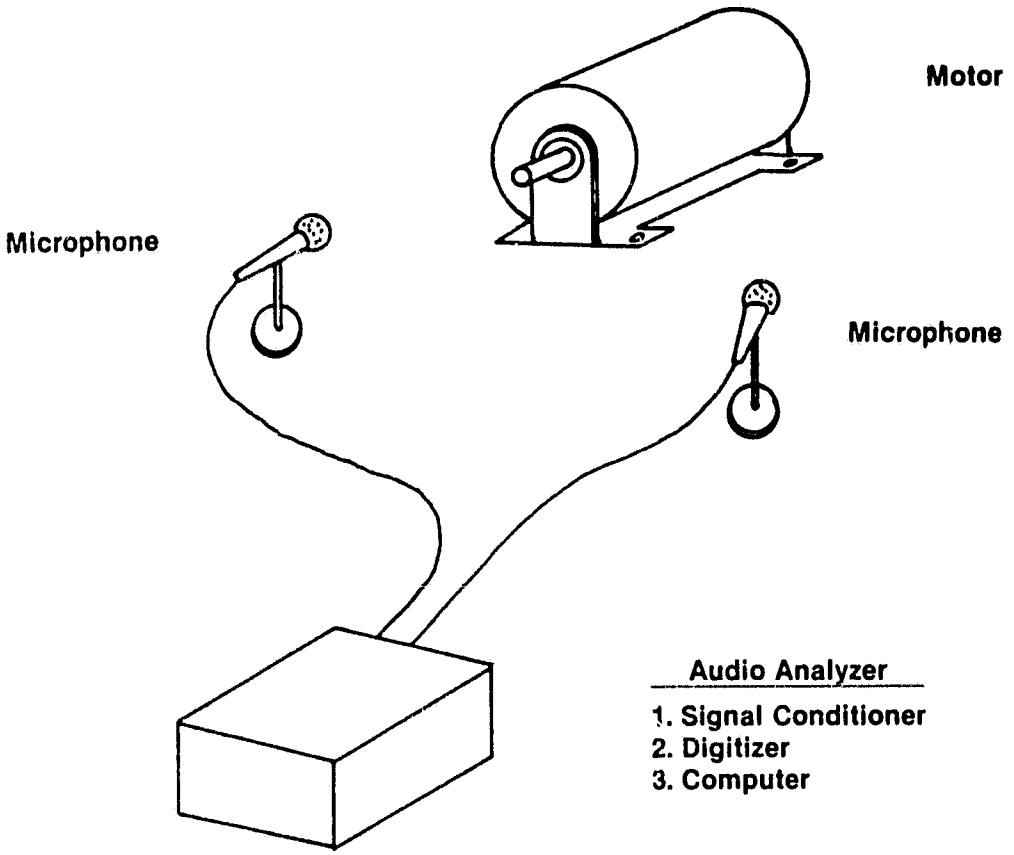


FIGURE 12 AN AUDIO ANALYZER

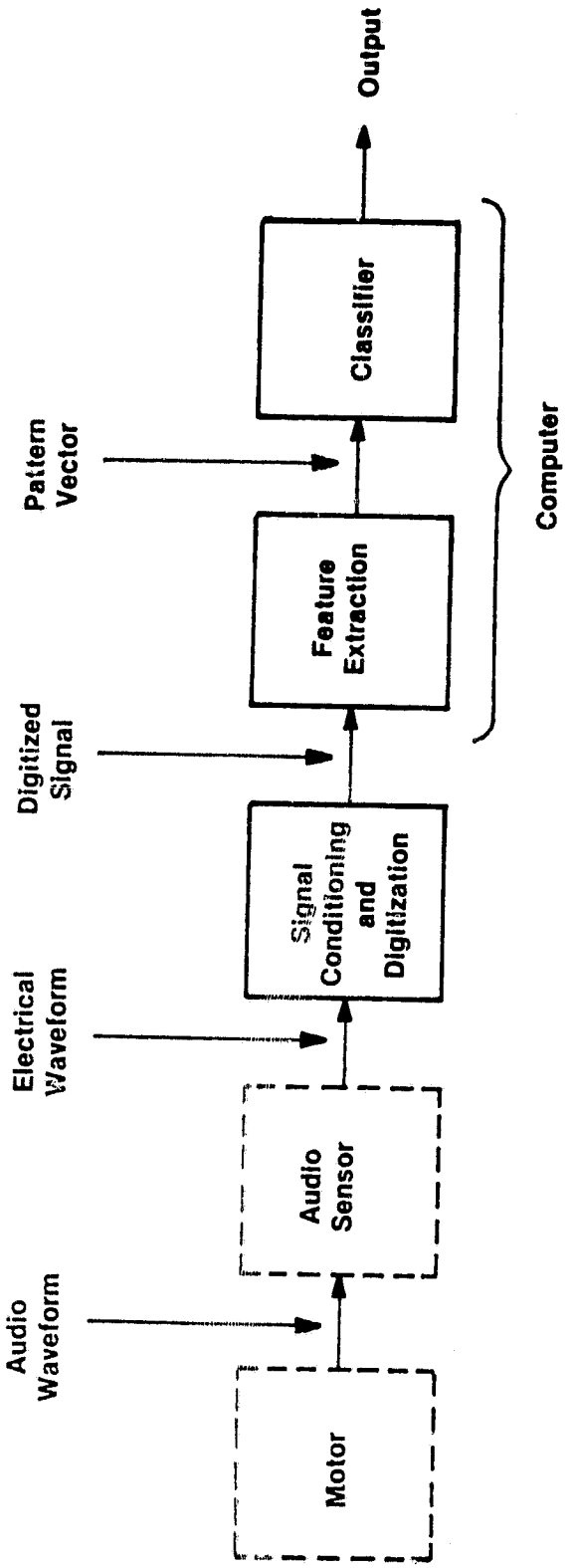


FIGURE 13 BLOCK DIAGRAM OF AUDIO ANALYZER

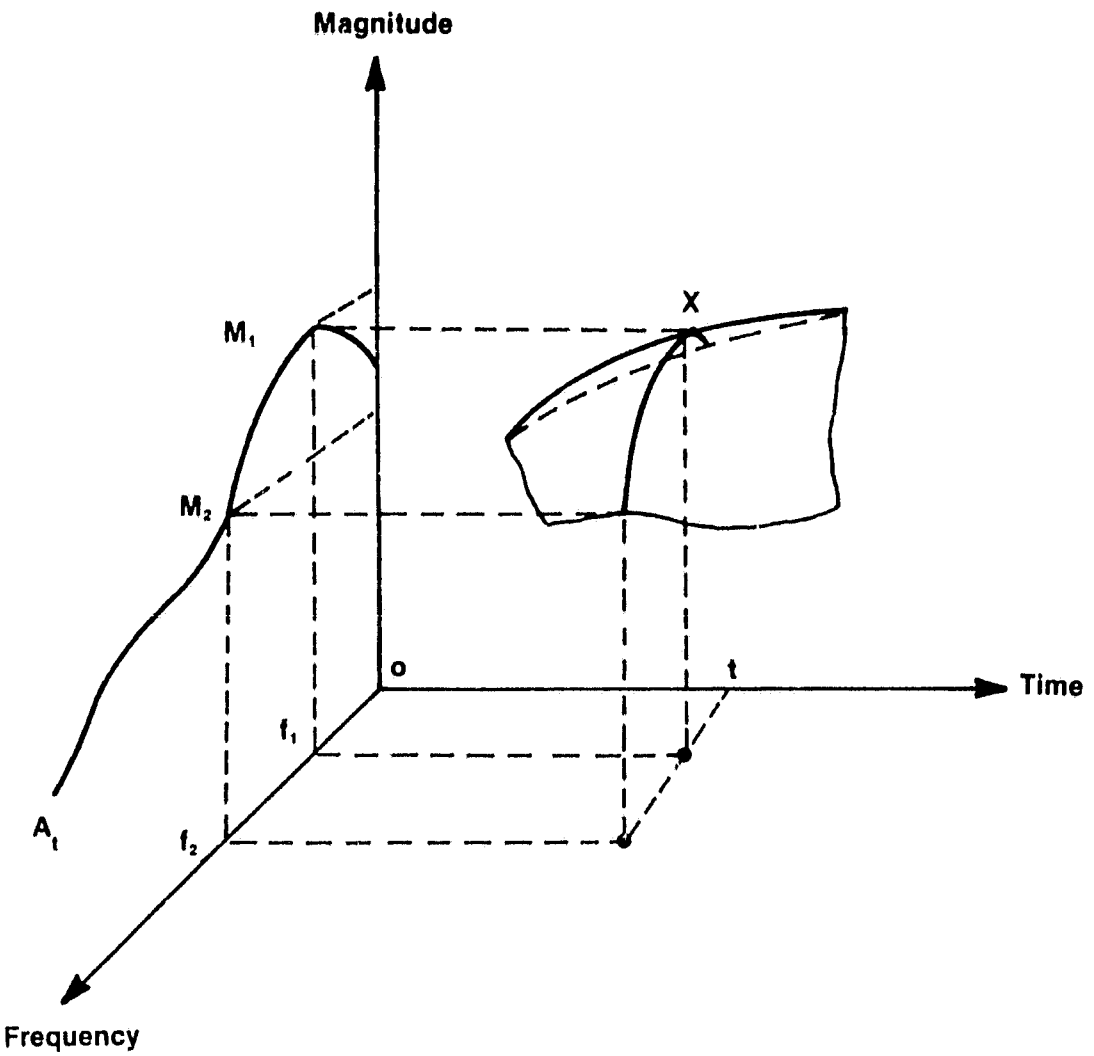
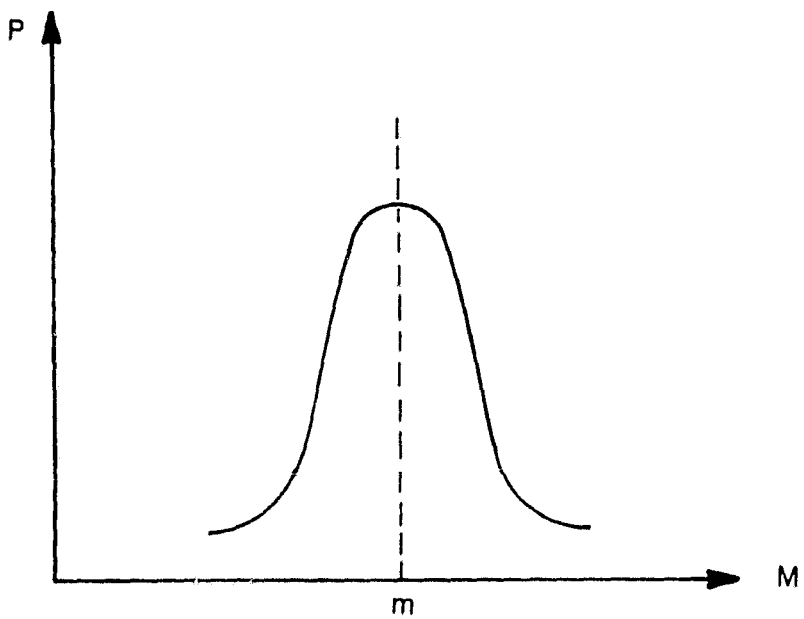


FIGURE 14 AN AUDIO PATTERN IN THREE-DIMENSIONAL SPACE



$$P(M) = \frac{1}{\delta\sqrt{2\pi}} \exp\left[-\frac{1}{2\delta^2}(M-m)^2\right]$$

δ = Deviation

m = Mean

FIGURE 15 ONE-DIMENSIONAL GAUSSIAN DISTRIBUTION

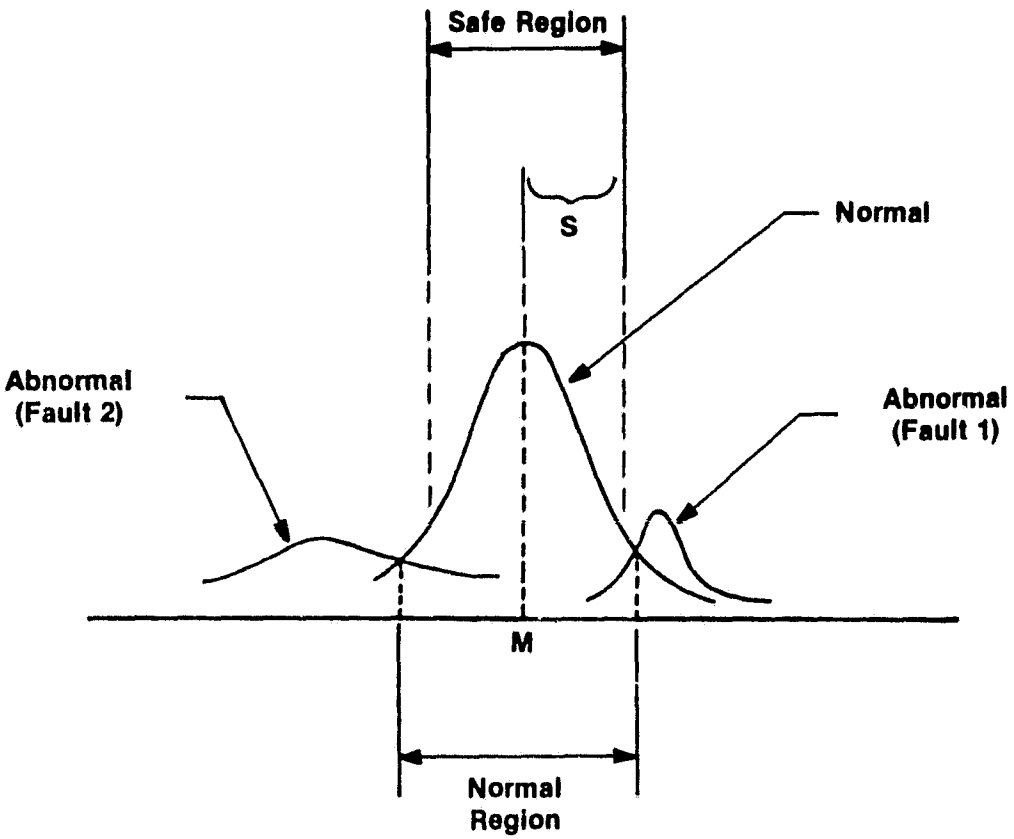


FIGURE 16 CLASSIFICATION OF PATTERNS

probability of the machine being normal is higher than being abnormal. For minimum risk, the computer is instructed to classify a pattern which falls in normal regions as normal and which falls outside normal regions as abnormal. This is called the Bayesian classification rule.

To simplify computation and to minimize memory size, a weighted distance method is suggested. In this method, when the weighted distance between the pattern and the center of distribution is greater than a safety distance S, the pattern is recognized as normal; otherwise, abnormal (see Figure 16). In other words, a safe region is made around the center of distribution of normal pattern. The pattern is considered normal and safe within this region. The decision rules can be described by the following inequalities:

$$\text{If} \quad \sum_{i=1}^{k} W_i(M_{tfi}-M_{fi})^2 \leq S^2, \text{ then the pattern is normal} \qquad (3)$$

otherwise, abnormal

where M_{fi} is the mean of variable M_{tfi}, see equation (2)

If the distributions of normal and abnormal patterns are far apart or the pattern distributions are special, the setpoint classification method may find possible use as the recognition scheme. The setpoint classification method is the method used in the present Model 100 C/M I. However, if the distribution functions of normal and abnormal patterns are Gaussian (normal distribution) and close to each other, the setpoint classification method will be inaccurate.

The recognition principles introduced in this section have been successfully used in various applications.[51,52]

It is often desired to know more about the abnormal patterns of a running machine. For instance, it is helpful to know the possible causes of the abnormality. The abnormal patterns can be divided into many classes with respect to the causes. Each class has a distribution function. Multiclass classification techniques[53] can then be applied.

Computer Learning. For the computer to operate as desired, it has to know where the safe region is or what the normal and abnormal parameters of the distribution function are. There are usually two ways of teaching the computers. One is the preset method and the other is by computer learning. In the preset method, the parameters are preset when the system is built. The computer learning capability allows the audio analyzer to infer the parameters by itself. Since the audio patterns are affected by the housing of the machine or the environment such as echo and other noises, the normal audio patterns of the same machine might be slightly different from system to system. The computer learning capability, thus, appeared to be more desirable. However, the available learning techniques[54] normally require long computation time and large memory.

A compromise is to use a development system to do the computer learning at the installation stage. Thereafter, the parameters are set for that particular running machine in that particular environment and the computer learning software is no longer needed in the audio analyzer memory. Such a special purpose signature analyzer can be called a noise watcher, which watches the

noise generated by the machine and provides warnings when the noise indicates that something is wrong.

The computer learning processes are usually categorized as sequential and nonsequential. The sequential procedures are more complex but require less memory for storing learning data. In the sequential procedure learning data are used and dropped one by one to upgrade the existing parameters. In the nonsequential procedure the learning data is collected, stored and processed at the same time. Although both methods will obtain the statistics from the learning samples, the sequential procedures fit our needs better since the computer can keep learning until the parameters are satisfactory without increasing the memory for learning samples.

Nonstationary Audio Pattern. In the previous discussion, the audio spectra of normal patterns are assumed to be stationary. This assumption reduces the features required to recognize a pattern and makes the analysis simpler. If a normal audio spectrum happens to be time-varying and repetitive, further research on sampling rate and periodicity is necessary. Also, the beginning sampling time should be studied to avoid the trouble caused by time shift. A periodic audio spectrum will inevitably require more features to characterize the pattern.

The time information is very important in performance trend analysis for fault prediction. As a simple example, if the noise of the running motor gets louder and louder, something must be going wrong with the motor.

Dimensionality. As mentioned before, the pattern is expressed by a vector which has many variables. Each of these variables is the loudness of the sound at a certain frequency. The number of variables required greatly affects the computation complexity and the memory requirement. Therefore, it is necessary to select some particular frequencies where the most important information can be extracted. The procedure is called feature selection.[55] Feature selection is very important to the performance of the audio signature analyzer. A feasibility study using existing selection techniques is necessary. If few variables are necessary to reach a very high recognition accuracy, a small sized computer will be sufficient to implement such a system. If a large number of features are required to reach a reasonable recognition accuracy, a larger computer may be needed. If the recognition accuracy cannot be met by audio sensors, more information provided by vibration sensors or some other invasive sensors may be necessary. In the worst case, the analyzer could be too complex to be feasible.

A feasibility study will require an audio sensor, a spectrum analyzer,[56] and several mechanical rotational machines of interest that are modifiable to produce abnormal patterns.

Calibration Curve Retention

When a sensor in a flight hardware C/M I is faulty it will be isolated and replaced with a spare sensor. The calibration curve of the spare sensor is not necessarily the same as the original one. The methods for and size impacts of retaining calibration curves in computer memory for spare sensors have been studied. The techniques of data compression and information coding are required.

Calibration Curve Storage Techniques. Four calibration curve storage techniques are presented in this section. In ideal cases the calibration curves can be represented by mathematical models ranging from straight lines to complex nonlinear parametric functions. When these mathematical models are not applicable, point-by-point storage can be used to represent the irregular curves. These methods are described in the following order of increasing complexity.

1. Linear Approximation. These calibration curves are represented by straight lines. This is the simplest method.

2. Nonlinear Parametric Approximation. The calibration curves are approximated by N-degree polynomial functions, exponential functions, or some other parametric functions in this method. The total number of parameters required is not large. The memory requirement will not be a problem.

3. Piecewise Linear Approximation. In this method each calibration curve is approximated by a number of linear functions, their coefficients are then stored in the memory for processing. This method can easily be applied to all kinds of curves, but the number of linear functions affects the approximation accuracy and the memory efficiency.

4. Point-By-Point Storage. With this method the calibration curves are stored point-by-point in the memory. This method requires more memory than those mentioned before. The interpolation method or weighted average method can be used to calculate the value between points.

   Different techniques are available for point-by-point storage of calibration curves. Fortunately, the sensor calibration curves normally do not require these techniques.

Storage Location Options. If the calibration curves can be approximated by functions with a small number of parameters it does not require a large memory space to store the parameters for the spare sensors. The calibration curve could be stored in the following locations:

1. Main Memory. The erasable/programmable read-only memory (EPROM) of the main computer memory can be used to store the curves.

2. Secondary Memory. For laboratory R&D C/M I's, the disk or cassette can be used to store a large amount of data. But they are not feasible for flight hardware applications because of the physical size. The bubble memory is a viable candidate for calibration curve retention.

3. At the Sensor Site. The calibration mechanism may be implemented as a part of the sensor with a microprocessor at the sensor site. In this approach the signal conditioning, digitization, multiplexing and calibration are distributed to the sensor sites. The signal transmitted to the computer will be digitized and calibrated.

The ARS Sensor Calibration Curves. The method used for storing a calibration curve depends on the characteristics of the calibration curve. In other words, the proper method of storing the calibration curve is sensor dependent.

The sensors used in the ARS hardware and the recommended storage techniques are shown in Table 4. Most of them are linearly calibrated within specified ranges. For the rest, second order polynomials can approximate the curves very well except for the flow rate sensor, which may need third order polynomials or exponential functions.

Because the linear, second and third order polynomial functions only require two to four parameters, the memory size impact is nominal. Storing them in the EPROM of the C/M I main computer memory is feasible. Some of the ARS sensors will be digitized and multiplexed at the sensor sites. In this case, the calibration curves will naturally be stored at the sensor sites.

Automatic In Situ Sensor Calibrations

The advantages and techniques of automatic in situ sensor calibrations were investigated. The sensor types used in an ARS are:

- Current
- Voltage
- Temperature
- Pressure
- Flow Rate
- Speed
- Dew Point
- Combustible Gas Concentration
- Relative Humidity

Advantages of In Situ Calibration. In an automated system such as the regenerative ARS, the system performance depends significantly on how well the sensors are calibrated. A sensor which is not calibrated may be hazardous to the system operation; the combustible gas concentration sensor is a good example. With in situ calibration capability, the sensors can be calibrated on-line frequently without the need for operator intervention. Automatic in situ calibration also avoids the operator errors implicit in a manual calibration. In summary, the advantages of automatic in situ calibration are:

- Calibration without crew service
- No human error
- More accurate sensors

Automatic In Situ Calibration Techniques. Automatic in situ calibrations typically require the generation of standard physical/chemical conditions for the sensor signal conditioner adjustments. For example, a standard pressure is required for zeroing the pressure sensor signal conditioner and another standard pressure is required for the adjustments of the signal conditioner span.

TABLE 4 RETENTION OF SPARE SENSOR CALIBRATION CURVES

A. Sensor Types and Calibration Curves

Sensor Type	Calibration Curve
Current	N/A (No Spares)
Voltage	N/A (No Spares)
Temperature	Linear
Pressure	Linear
Flow Rate	Second or Third Order Poly.
Speed	Linear
Dew Point	Linear
Combustible Gas	Second Order Poly.
RH	N/A (Calculated, No Spares)

B. Storage Location Options

- Main Memory
- Secondary Memory
- Part of the Sensor

Generation of the standard physical/chemical calibration conditions is unique to each sensor type. The calibration sequences and the techniques after the standard conditions have been generated are common for all types of sensors. The common sequence of events in an automatic in situ calibration function is as follows:

1. Generation of the standard sensor condition for signal conditioner zero adjustment.

2. Adjustments of the electronic zero circuit in the sensor signal conditioner.

3. Generation of the sensor condition for span adjustment.

4. Adjustment of the electronic span circuit in the sensor signal conditioner.

5. Sensor calibration completed; return to normal operation.

The sequence of events described above is only applicable to the sensors which have linear calibration curves within the application range. For the sensors with nonlinear calibration curves within the application range, more than two calibration points are required and the automatic in situ calibration capability should include the generation of multiple calibration conditions. For the regenerative ARS application, however, the linear approximation of sensor curves is acceptable. This means that only two calibration points are required during the automatic in situ calibration operation. The sequence of events described above may have to be repeated depending on (1) how critical the sensor is, and (2) how fast the sensor drifts. In general, it is adequate to repeat the calibration sequence once. That is, the sequence "put sensor off line, generate zero condition, adjust zero, generate span condition, adjust span, generate zero condition, adjust zero, generate span condition, adjust span and put sensor back on line" is adequate for most of the regenerative ARS sensors automatic in situ calibration.

Automatic In Situ Generation of Calibration Conditions. To perform automatic in situ calibration, the calibration conditions such as reference voltage, standard temperature, standard pressure and standard combustible gas concentration are required. For some of the sensors, generation of the calibration conditions may be too difficult to be feasible.

Table 5 summarizes the readily available calibration techniques for ARS application. Current and voltage sensors can be calibrated by using standard voltage signals generated by using zener reference diodes. Automatic in situ calibration of speed sensors can be achieved by the generation of electrical pulses which simulate the optical/magnetic pickups from a speed sensor. Combustible gas concentration sensors can be calibrated using water vapor electrolysis (WVE) generated hydrogen (H_2) concentration in air.[57]

Temperature, pressure, flow rate, dew point and relative humidity are difficult, without further development efforts, to calibrate with automatic in situ calibration functions. For the critical sensors in a system, the alternatives to avoid technical difficulties in automatic in situ calibrations are:

TABLE 5 AUTOMATIC IN SITU SENSOR CALIBRATION

Sensor Type	Feasible Calibration Technique	Alternative
Current	Standard Reference Voltage	----
Voltage	Standard Reference Voltage	----
Temperature	----	Triple Redundant or Spare
Pressure	----	Triple Redundant or Spare
Flow Rate	----	Triple Redundant or Spare
Speed	Electrical Pulse Generator	----
Dew Point	----	Triple Redundant or Spare
Combustible Gas	Water Vapor Electrolysis	----
RH	----	Triple Redundant or Spare

a. Use triple redundant sensors.
b. Use a spare sensor with calibration curve stored in the computer memory.

System Reliability and Maintainability Impact. Automatic in situ sensor calibration capability can increase system reliability and maintainability. The capability increases system reliability by avoiding shutdowns caused by uncalibrated sensors. It increases the maintainability by the reduction or elimination of operator intervention during a calibration procedure.

On-Line Real-Time Diagnosis

A diagnostic program can be used by the C/M I computer to find out the cause of an out-of-tolerance event.

When an out-of-tolerance event occurs, the sensor values before, during and after the event usually show some implications to the cause of the event. Very often, an on-line real-time diagnostic is desired. This section discusses the concept and feasibility of implementing such a system.

Approach 1. When an out-of-tolerance event occurs, the computer automatically calls the diagnostic program which will then search for useful information through a certain period of sensor data history which has been stored in the computer memory. This approach requires a large memory for storing the sensor data. Some of the sensor data may not be useful for diagnosing that particular event. Also, the result of this diagnostic program may be too late to prevent the system from shutdown because the execution of such a diagnostic program requires time. This approach has an advantage, however, in that it does not require CPU time if no out-of-tolerance event occurs.

Approach 2. The diagnostic program runs all the time and keeps track of the system running conditions. The system performing conditions can be divided into many stages, each stage is represented by a "state."[58] The event implications detected from the sensor data are described in the state. In other words, the diagnostic program is processing the sensor data all the time and may send out early warning messages when the system enters certain states. This approach has two advantages. One, it does not require a large memory size. The diagnostic program processes the data and drops out the useless data, extracts the important information and updates the present state. The sensor data history does not have to be stored in the memory. Two, the diagnostic result is available whenever the event occurs unless the diagnostic requires some post-event information. This approach requires CPU time when the system is running. If the CPU has enough free time to run the diagnostic program, this requirement has no negative effect on the system. It becomes a drawback, however, when the CPU time is not sufficient to run the diagnostic program.

Comparing the above two approaches, the second one is better than the first one if the CPU is not overloaded by the diagnostic program. The efficiency and accuracy of such a diagnostic program is strongly related with the nature and characteristics of the mechanical system. If the implication of the event is very complicated and ambiguous, an accurate diagnostic program will be very difficult to implement. A diagnostic program itself may occupy a large memory

space and require long computation time. If the implication of events is simple, clear and easy to be divided into a small number of states, the diagnostic program implemented will be efficient and accurate.

The following procedure describes how to implement such a diagnostic program using Approach 2:

1. The nature and characteristics of the parts and function of the mechanical system should be fully understood. The implication in the sensor data of the time proceeding and following any out-of-tolerance event and the corresponding causes should be analyzed and fully understood.

2. The implication should be studied in detail and divided into representative states. Some descriptive variables (attributes) may be necessary to describe the state. The interrelationship among the states should also be defined. For example, M_x may represent the current monotonously increasing at rate x, $A_{p,x}$ may represent a voltage changing up and down alternately with period p and present changing rate x.

3. Prepare the state transition diagram according to the interrelationship among the states. The computation of attributes when a transition is made should also be formulated. For example, Figure 17 shows a state transition diagram. Each circle represents a state. The double circle indicates the normal state. An input is represented by an "i." Statements in lower half of the circles are the warning messages output by the computer at that state. The conditions for a transition and the computation rules of the attributes are given in Table 6. Figure 18 represents some possible sensor value curves. With the proposed diagnostic program, the curves are already characterized by the states when warning occurs. Storage for the past 16 sets of sensor data is no longer necessary.

4. Write a diagnostic program that simulates the state transition diagram. Such an algorithm is usually referred to as an "attributed automaton"[59,60] in Formal Languages and/or Artificial Intelligence. Using a high level language to implement such a software routine is recommended because of the simplicity of modifying the transition diagram.

Approach 2 requires less memory for data storage than Approach 1 and may give warning early enough to prevent a hardware failure. The accuracy of diagnosing results depends on the completeness of the transition diagram. The complexity of the transition diagram determines memory size requirements and software implementation effort.

DEMONSTRATION

As a part of the program effort, the design and demonstration of advanced fault diagnostic instrumentation concepts were completed. These concepts can become

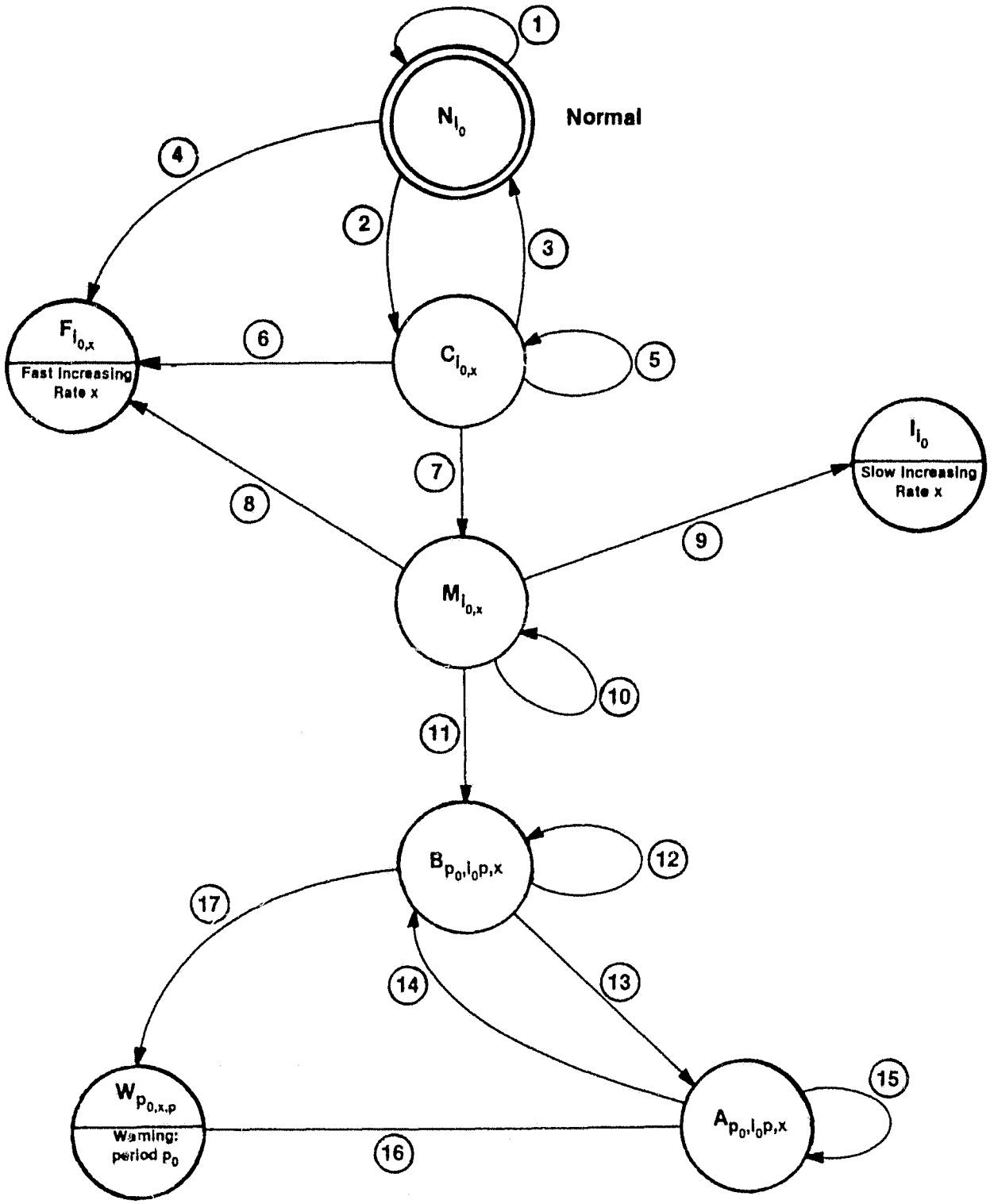


FIGURE 17 A STATE TRANSITION DIAGRAM

TABLE 6 TRANSITION CONDITIONS AND ATTRIBUTE RULES

Transition	Conditions	Attribute Computation Rules
1	$i_l < i < i_{hl}$	$i_0 = i$
2	$i \geq i_{hl}$	$x = i - i_0$ $i_0 = i$
3	$i_l < i < i_{hl}$	$x = i - i_0$ $i_0 = i$
4	$i - i_0 > i_{h3} - i_{hl}$	$x = i - i_0$ $i_0 = i$
5	$i_{h2} > i - > i_{hl}$	$x = i - i_0$ $i_0 = i$
6	$i \geq i_{h3}$	$x = i - i_0$ $i_0 = i$
7	$i_{h3} > i > i_{h2}$	$x = i - i_0$ $i_0 = i$
8	$i - i_0 > x_h$	$x = i - i_0$ $i_0 = i$
9	$i > i_{h3}$	$i_0 = i$
10	$i_{h3} > i > i_0 > i_{h2}$	$x = i - i_0$ $i_0 = 1$
11	$i < i_{h2}$	$p_0 = 0$ $p = 1$ $x = i - i_0$ $i_0 = i$
12	$i < i_{h2}$	$p = p + 1$, $x = i - i_0$ $p_0 = p$, $i_0 = i$
13	$i_{h3} > i > i_{h2}$	$p = p + 1$ $x = i - i_0$ $i_0 = i$
14	$i_{h2} > i$	$p_0 = p$ or $(p + p_0)^{\frac{1}{2}}$ $p = 1$ $x = i - i_0$ $i_0 = i$

continued-

Table 6 - continued

Transition	Conditions	Attribute Computation Rules
15	$i_{h3} > i > i_{h2}$	$p = p + 1$ $x = i - i_0$ $i_0 = i$
16	$i > i_{h3}$	$p_0 = p$ or $(p + p_0)^{\frac{1}{2}}$ $x = i - i_0$, $i_0 = i$
17	$i > i_{h3}$	$p_0 = p$ or $(p + 1 + p_0)^{\frac{1}{2}}$ $x = i - i_0$

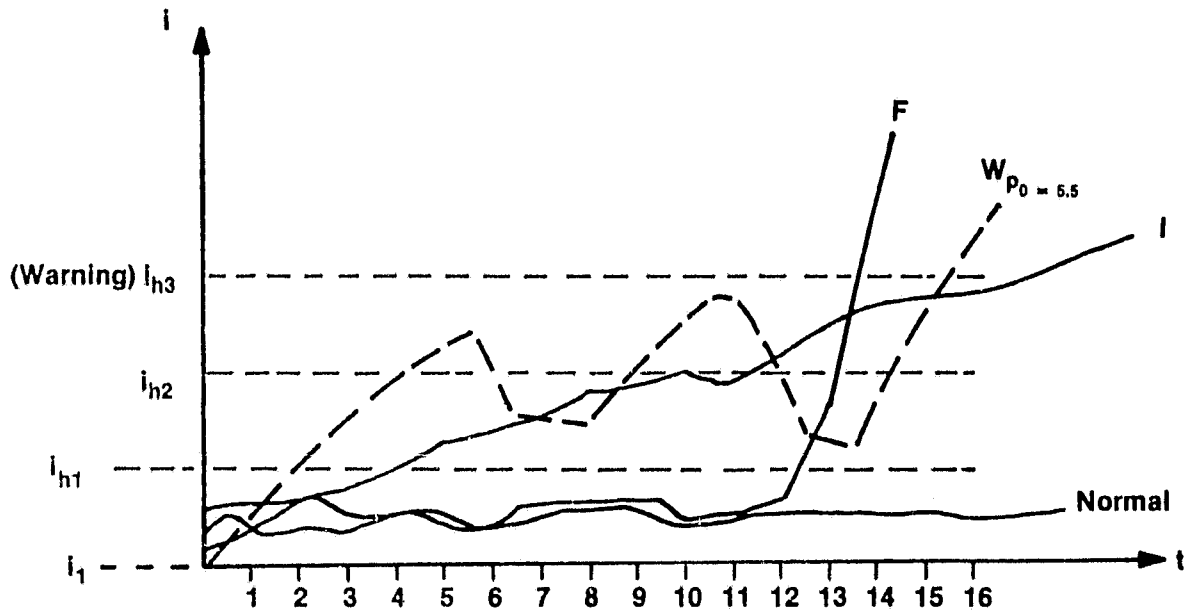


FIGURE 18 SOME POSSIBLE SENSOR VALUE CURVES

standards for the overall EC/LSS hardware development. Two specific concepts were developed to the stage of laboratory demonstration. They are: (a) intercomputer communication protocols with error detection capability and (b) microprocessor-based BID circuit.

Intercomputer Communication with Error Detection

Two intercomputer communication protocols were established for communication between the computers. These computer links render a Computer Automation LSI-2 minicomputer capable of communicating with another LSI-2 minicomputer or an Intel 8-bit microcomputer through the standard Electronic Industries Association (EIA) RS-232C interface (an industry standard bit-serial digital transmission protocol) with transmission error checking capability. A typical application of the computer links is the connection between subsystem instrumentation and the central computer. Another example of application is the connection of an instrumentation prototype to a DARS during the R&D phase of the development. This connection is presently done with a pair of byte-parallel general purpose picoprocessors, for which the cable length is limited to 15 feet. Using the RS-232C link, the picoprocessors are eliminated and the cable length is increased to 100 feet.

Link 1: Between LSI-2 Computers

For two computers to talk with each other through the EIA RS-232C port, their selection of parity checking, stop bit, character length, and the transmission baud rate (bits/second) should be the same. The communication link for the two LSI-2 computers requires a cable of five conductors: the signal received (EIAR), transmitted (EIAT), Request to Send (RTS), Clear to Send (CTS) and a common ground between the two interfaces. The protocol designed requires that the interface time delay circuit be disabled.[61]

The transmission can be done by a simple program as shown in Figure 19. Using these instructions, however, the receiver should always be active prior to the transmitter. To avoid this timing problem, the handshaking instructions between the two computers and the time delay instructions in the transmitter can be added before enabling the interrupt. The receiver outputs a logic one to the RTS line (CTS line of transmitter) and then waits for the buffer to be filled with input information. The transmitter keeps waiting and sensing until CTS is logic one and then starts the output sequence. After this handshaking the transmitter uses a counter to delay itself several instructions in order to ensure that the receiver becomes active first and then starts automatic I/O. At the end of the automatic I/O operation a checksum of the transmitted data is generated by the receiver and compared to one generated and transmitted by the transmitter. This protocol is shown in the flow charts in Figure 20. The checksum represents the sum of all the transferred words with the carries ignored. The checksum is used to detect the transmission error. Using this protocol, two computers can communicate with each other while both systems are running. Before the transmission begins the checksum is calculated by the transmitter and transferred to the receiver as two bytes during handshaking. At the end of the automatic I/O the system will call the End of Block (EOB) subroutines. In the EOB subroutines the byte counts are checked at both sites and the checksum is calculated at the receiver site and compared with the transferred checksum to detect transmission errors. If any error is detected

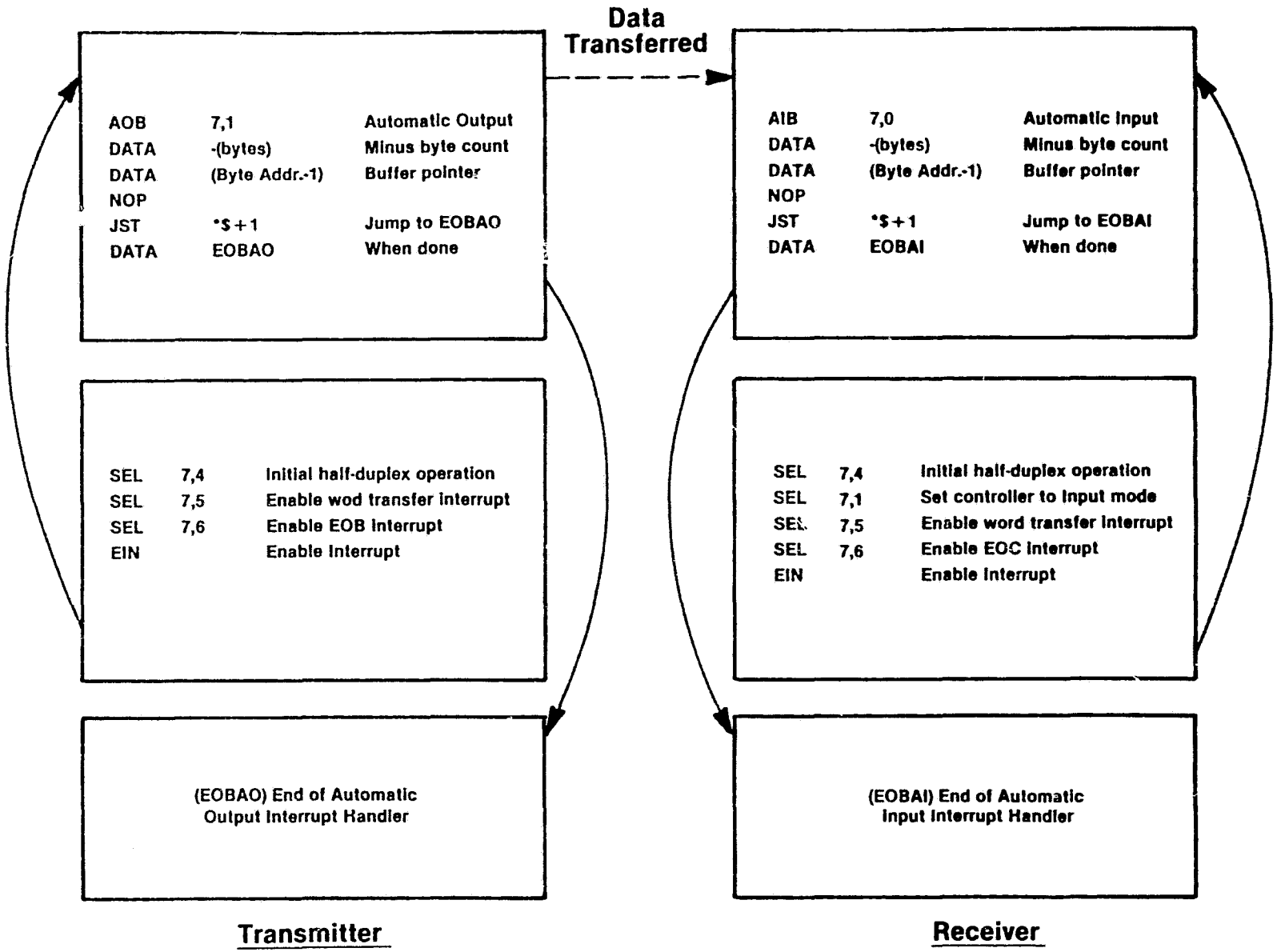


FIGURE 19 A SIMPLE COMPUTER COMMUNICATION PROGRAM

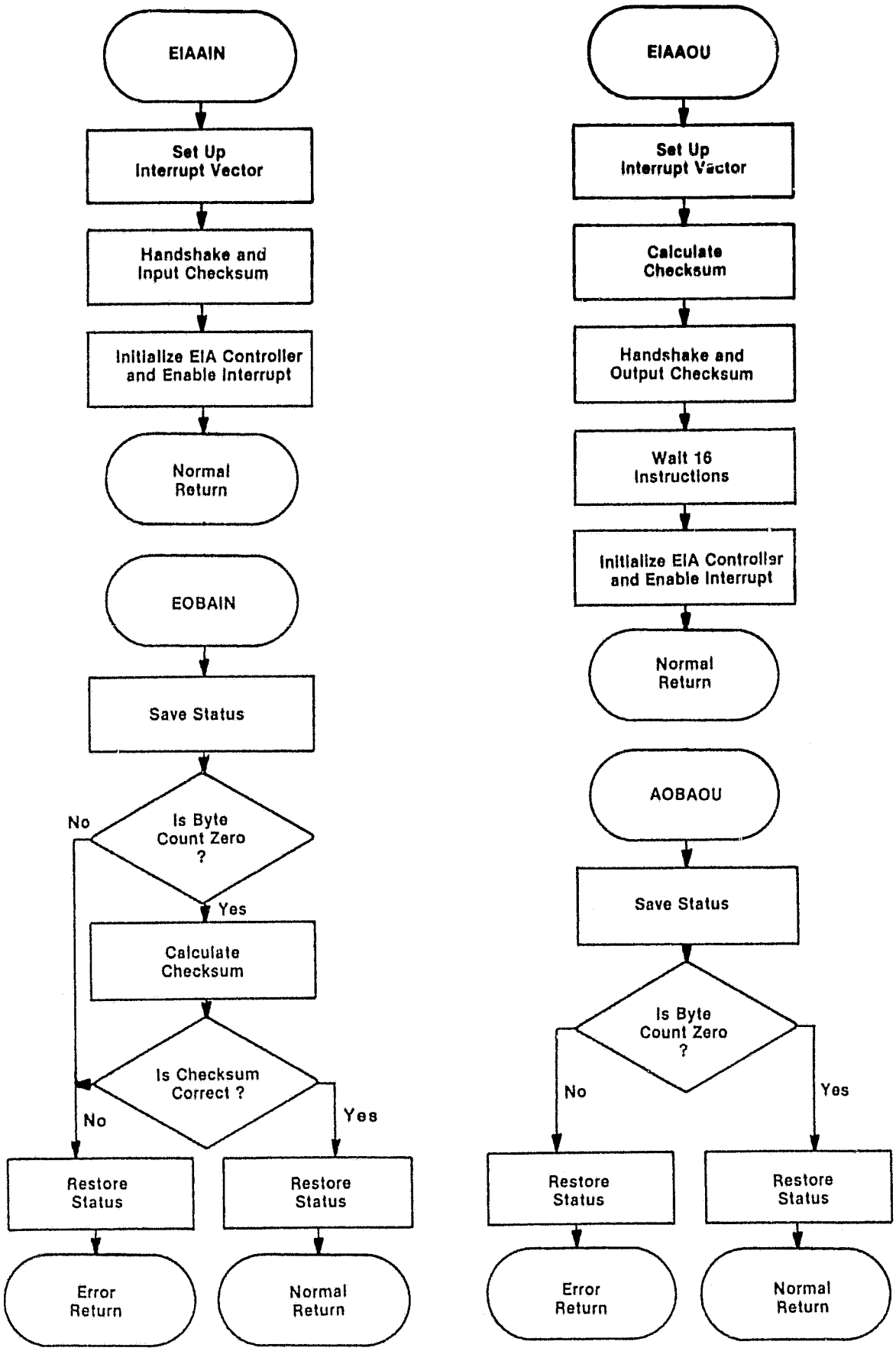


FIGURE 20 TRANSMISSION PROTOCOL FLOW CHART

it will jump to the error return location. The checksum calculation, storage and transmission are transparent to the user. The user only needs to supply the minus byte count and the byte address of the I/O buffer minus one, and two locations for normal and error returns. From the error return, the user may discard the transferred data and request another transfer.

Link 2: Between LSI-2 and Intel 8085 Microcomputer

This link is designed to connect the EIA RS-232C port of the LSI-2 minicomputer and the CRT port of the Intel microcomputer development system (MDS). The Intel MDS is wired so that the CTS and RTS are not accessible from the CRT port. Therefore the Data Set Ready (DSR) and Data Terminal Ready (DTR) signals are used instead. As mentioned previously, the selections of parity check, stop bit, character length and transmission baud rate should be the same for two computers to communicate through the EIA port. The Intel MDS can set/reset the values of DSR and DTR. It is the user's responsibility to set the signals of DSR and DTR to zero at the microcomputer site before and after the transfer. The handshaking of the two computers is started from the MDS site. After handshaking the transmission starts and transfers information byte by byte. At the transmitter site the checksum byte is calculated before the transmission and transferred as the last byte. The receiver will receive all the information bytes and the checksum byte, and then will calculate the checksum and compares the results. The LSI-2 will jump to the error return location if a transmission error is detected after receiving data. The microcomputer will have the accumulator equal to one when returning to caller routine if error is detected after receiving data. The checksum operation is transparent to users. The user should provide the negative byte count and byte address of the I/O buffer. At the LSI-2 site the user should also provide two return locations. At the Intel MDS site the information should be passed to the I/O routines by registers; it is the user's responsibility to check the accumulator for error information after receiving data.

The described two links not only can transfer data between computers but also detect the transmission error. The byte count checking and the checksum operation programmed in the I/O subroutines detect the transmission errors. These operations are transparent to users and assure the transmission accuracy.

Built-in Diagnostic Circuit

An Intel 8085 microprocessor-based circuit is designed to perform BID functions for a microcomputer or minicomputer implemented C/M I. The BID circuit has the following functions:

- Requests the C/M I computer to initiate its Self-Diagnostic Program (SDP) which verifies the functions of the CPU, memory and data bus.
- Interprets the SDP results as transmitted from the C/M I to the BID circuit.
- Monitors the A/D interface card function.

- Monitors the C/M I computer to verify that it is active.
- Monitors, as a redundant fault detection function, up to two critical process parameters directly (bypassing the computer and A/D interface).
- Outputs an emergency shutdown request signal when critical malfunctions have been detected.

The block diagram of the BID circuit was shown previously in Figure 10. Figure 21 shows the actual circuit card layout of the BID.

Hardware Design

The BID circuit, as shown in Figure 22, was designed with state-of-the-art microprocessor technology. The heart of the BID is an Intel 8085 microprocessor which is supported by an Intel 8155 RAM/IO (Random Access Memory/Input-Output), a 2716 EPROM and an 8212 I/O. With this electronic configuration, the BID has a memory capacity of 2K bytes of EPROM and 256 bytes of RAM. It has 22 general purpose I/O lines divided into three ports. A comparator logic is used to monitor the critical process parameters. The two parameters are compared with high and low setpoints which are set at the absolute tolerance levels. The setpoints are adjustable by on-card potentiometers. The results of the comparator logic are sent to the microprocessor for a diagnostic decision.

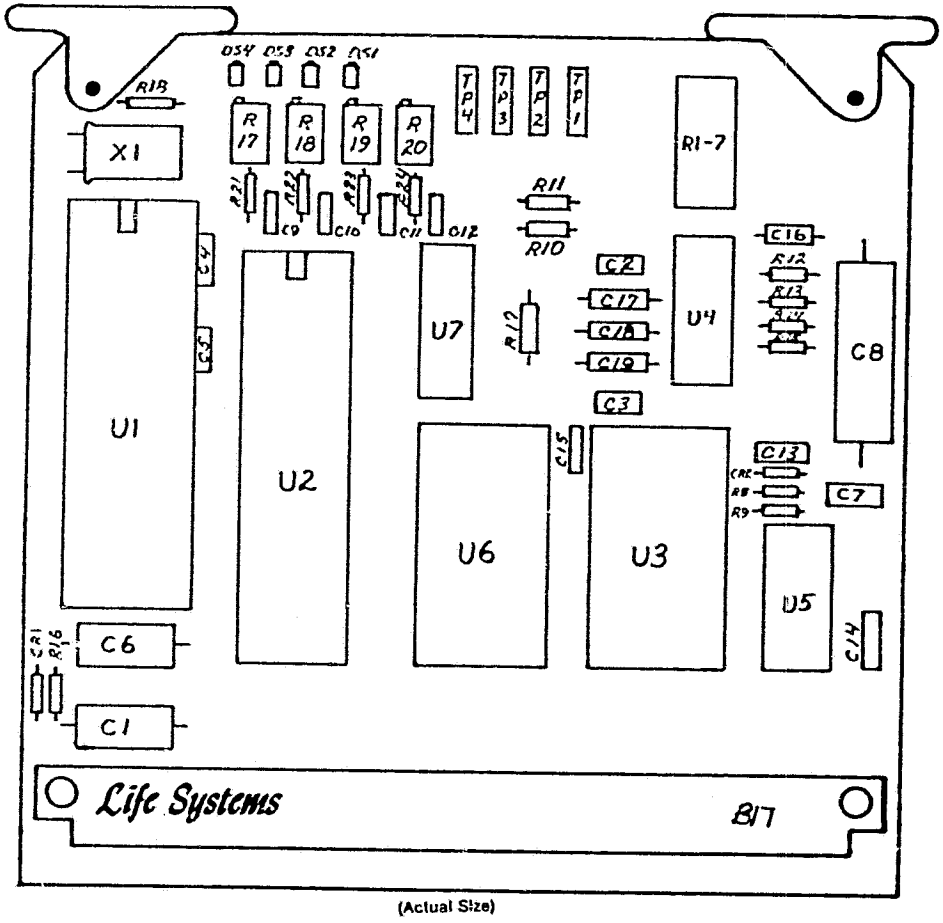
To detect malfunctions of the CPU and A/D interface of the C/M I, the memory reference control (read or write) signals and a software controlled A/D interface pulse train are checked by the BID circuit. The checking is accomplished by using retriggerable multivibrators. The retriggerable multivibrators are triggered at a constant frequency when the C/M I and A/D interface are functioning properly. The pulse width of the one-shot are designed to be greater than the triggering periods. The one-shot outputs should always be at the triggered state until the pulses stop. The microprocessor will monitor the output levels which indicate whether the C/M I computer or A/D interface card are functioning properly or not.

Software Design

The BID software program consists of the following:

- Power-up Routine
- Timer Interrupt Routine
- Monitor A which checks the digital input signals to verify that the critical process parameters are within tolerance range, C/M I computers are running and A/D interface is outputting pulses
- Monitor B which signals the C/M I computer for the initiation of the SDP inside the C/M I and interprets the SDP results

The functions which Monitor A performs are designed to ensure that the C/M I computer is running, the computer can output valid data through the A/D inter-



- State-of-the-art microprocessor technology
- Detects C/M I Computer CPU, memory, data bus, software and analog/digital (A/D) interface failures
- Performs fail-safe supplementary shutdown controller functions
- Compatible with C/M I 100 and C/M I 200

FIGURE 21 MICROPROCESSOR-BASED BID CIRCUIT CARD

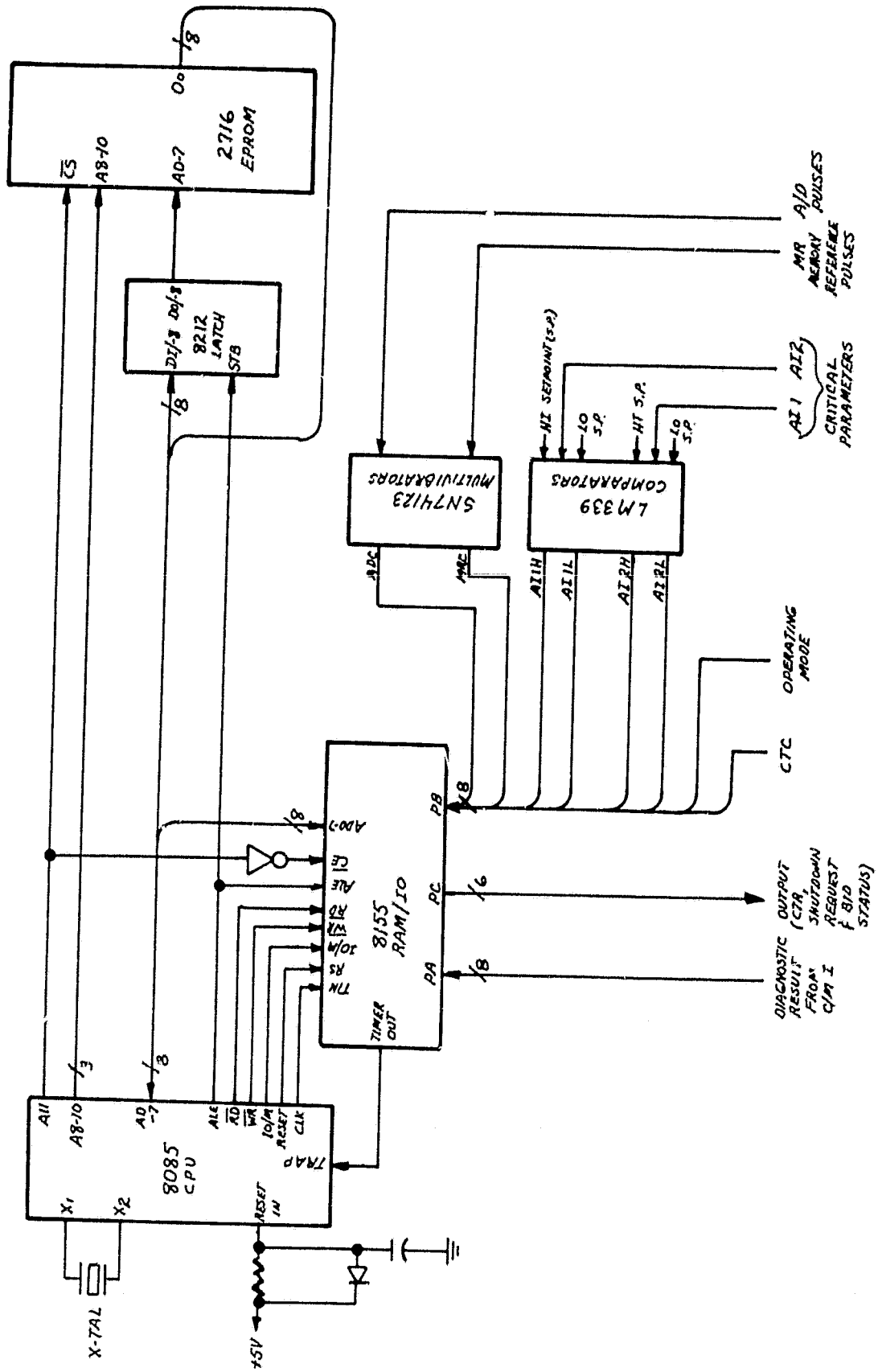


FIGURE 22 BID CIRCUIT SCHEMATIC

face and critical process parameters are within the safety range. If any of the above is not true the BID circuit will request an emergency shutdown. In some cases the emergency shutdown sequence may be a one step "power off" operation. In others, the sequence may consist of actuators off, purging and placing valves to shutdown positions. For the latter case, an emergency shutdown controller activated by the BID and other diagnostic/checkout circuits should be incorporated.

Monitor B is a diagnostic program which requests the C/M I computer to start self-diagnosis by outputting a request signal (logic one) called Computer Test Request (CTR). The C/M I computer, upon receipt of the CTR signal, will start a self-diagnostic sequence to check out the functions of the CPU, memory and data bus. The SDP program inside the C/M I is driven by the C/M I Real Time Executive (RTE) program. It runs through a computer checkout routine and, at the end of the checkout, outputs a 7-bit result to the BID. The SDP consists of four steps:

a. Buffer and bus line checkout. Different bit patterns are used to check whether the address, data or control lines are stuck at a logic one or logic zero.

b. CPU checkout. Starts with a specific data byte and runs through major instructions, such as load, add, shift, complement and store, to obtain a result data byte. This result byte will then be sent back to the BID for interpretation and diagnosis. Only the BID can make a judgment of whether the C/M I has successfully passed the self-diagnostic procedure.

c. EPROM checkout. Uses checksum method to check out the EPROM memory. A checksum word (or byte) was previously generated and stored for a block of EPROM at the time the program was created. To verify that the CPU and EPROM are functioning properly the checksum is regenerated at run-time and compared with the stored value. The newly generated checksum should always be identical to the previously generated one or a computer or memory failure has occurred.

d. RAM checkout. Checks out the RAM by using the sequence of read and store the data, write a specific data, read and check the written data and then store the original data.

When an error is detected in the four steps a coded bit pattern will be sent to the BID to indicate the cause of the failure. Otherwise, the result byte of step (b) will be sent to the BID. A Computer Test Completed (CTC) signal is sent from the C/M I (logic one means test completed) to the BID at the same time. When the BID concludes that the C/M I has had a malfunction it will request an emergency shutdown.

Laboratory Demonstration Observation

The demonstrated communication protocol allows reliable data/information to be transferred between computers. A secondary benefit of this demonstration is

the readiness of the bit-serial RS-232C communication design for central/subsystem and subsystem/DARS interface. The previously designed byte-parallel protocol is a simpler operation transferring parallel signals over a 48-conductor ribbon cable which has a length limitation of 15 feet. The bit-serial RS-232C protocol uses a cable of five conductors and has a longer allowable length of 100 feet. The communication speed of the serial protocol, although slower than that of the parallel protocol, is still well within the EC/LSS application requirement. For example, in EC/LSS DARS application the speed requirement is one data set (64 points) per two minutes. The projected subsystem to central communication speed will be 64 bytes per second (based on the assumption that the central system will respond to a subsystem request within one second). The demonstrated 1,200 baud (bits/second) nominal speed and 9,600 baud maximum speed are more than twice and 18 times that required, respectively.

The microprocessor-based BID circuit was demonstrated successfully with the Intel MDS In-Circuit Emulator (ICE). Some problems associated with erroneous data/control bus signals were observed when running with the actual 8085 CPU. The noises were attributed to crosstalk of the breadboard circuit wires. Elimination of the difficulties is projected when the BID circuit is printed on a printed circuit (PC) card.

CONCLUSIONS

The following conclusions are a direct result of this instrumentation development program:

1. Fault tolerance can be achieved by transmission error checking with software recovery algorithms, redundant components with comparison logic and redundant components with BID/BIC circuits.

2. Reliability of a system with voting and spares within the system is inherently higher than that of a group of four redundant systems with external voting logic.

3. A microprocessor-based BID circuit card having dimensions of 11.4 cm x 11.2 cm (4.5 in x 4.4 in) is feasible. It can detect failures of CPU, memory, data bus, A/D interface and software in addition to monitoring critical process parameters.

4. The technology of vibration signature analysis for actuator fault detection and prediction is available but the difficulty of vibration sensor installation is a road block. Audio signature analysis instead of vibration signature analysis is more suitable for EC/LSS applications. More work is needed to engineer the audio signature analyzer for EC/LSS application.

5. Most EC/LSS sensor calibration curves are linear straight lines (e.g., temperature, pressure, speed and dew point) or second/third order polynominal approximation (e.g., flow rate and combustible gas concentration). Storage of spare sensor calibration curves is, thus, not a difficult task. The retrieval of these calibration curves is more difficult. The ultimate answer to calibration curve

retention and retrieval for spare sensors is to incorporate them at the sensor heads.

6. Automatic in situ sensor calibration techniques are readily applicable to current, voltage, speed and combustible gas concentration sensors. Its application to temperature, pressure, flow rate, dew point and relative humidity sensors is difficult without further work on in situ calibration condition generation. The alternatives are triple redundant sensors (high reliability) or spare sensors with built-in calibration curves (high maintainability).

7. Recording and storing data for fault isolation requires a large size memory and is thus not feasible. This problem can be avoided by using the state transition technique. In this technique different states leading to a failure are defined and events that cause the system to go from one state to another are identified as attributes. Only current state and attributes are stored. Thus, real-time diagnosis could be done with less memory requirement. The state transition technique also provides fault prediction capability in addition to fault isolation.

RECOMMENDATIONS

Additional development efforts should be directed towards selection and incorporation of fault diagnostic functions available to EC/LSS application. The intercomputer communication with error checking capability developed under this program should be used in program loading and future central/subsystem instrumentation links. The BID circuit should be used in the future C/M I 100A for fail-safe operation and in the C/M I 200 for fail-safe and ultimately fault tolerant operation. The design of the C/M I 200 should have the following sequence:

1. Minimum Model 200 C/M I definition. Define the minimum C/M I needed to perform the process function on a continuous basis. The definition should include the desired availability expressed in terms of MTBF and MTTR for the process and the major subassemblies.

2. Capabilities versus penalties trade-off analysis. Any additional instrumentation hardware shall be quantified in terms of equivalent weight penalties. Only the added capabilities which can be justified from a size viewpoint should be included. The design must provide for fail-safe process operation. The process subsystem should be designed for an availability level reflected in operating for 90 days with three shutdowns, each being able to be repaired within two hours. The C/M I for the process subsystem should be designed to have no shutdowns for the 90 days of operation with replacement and maintenance allowed during the three shutdowns caused by process malfunctions. Flexibility of C/M I designed for the C/M I 100 series should be eliminated for reduction of size. Equivalent weight penalties associated with spacecraft power and heat rejection will be assumed as follows: 0.27 kg/W (0.59 lb/W) for DC power, 0.32 kg/W (0.71 lb/W) for AC power, 0.08 kg/W (0.184 lb/W) for heat rejection to liquid coolant and 0.20 kg/W (0.437 lb/W) for heat rejection to air.

3. Design of advanced C/M I Model 200. This effort shall establish projected space flight mission criteria, with primary emphasis on mission length and quantified reliability requirements. Based on these numbers, failure rates for each of the LRUs in the subsystem should be established. The number of spares or redundancies should be defined to meet the reliability goal for a given time period. The added components, such as sensors and circuits within the C/M I to allow fault diagnostic functions, should be defined in order to meet the availability goal of the total subsystem. The activity will result in a C/M I package ready for approval by NASA.

4. Instrumentation mock-up effort. The mock-up of the flight instrumentation design should answer the following questions:

   a. Which part of the C/M I should be removed from the instrumentation enclosure and packaged closer to or integrated with the mechanical electrochemical assemblies?

   b. Which, if any, capability shall be packaged as a separate, in-flight diagnostic unit? This is to be differentiated from the more conventional TSA or flight-related ground support accessories.

   c. Will the power supply of the C/M I be maintainable at the subassembly or the component level?

   d. Can the power supply be fractionalized so that several power supplies exist for the subsystem?

   e. What steps are needed to build the custom large-scale integrated circuits to allow incorporation of the above conceived capabilities into the future hardware at minimum penalty in reliability, size and estimated cost?

   f. What is the operator/system interface and subsystem/central interface projected for the flight hardware?

   g. What aspects of the developer's knowledge are projected to be incorporated into the flight hardware?

   h. Will multiplexing of signal conditioning be advantageous and feasible?

5. Further advanced development. In future development efforts, a program is needed to lay out the routes for obtaining custom large-scale integrated circuits for EC/LSS applications. Selecting, designing and obtaining custom large-scale integrated circuits comes later when the flight mission is ready. Other development efforts should include the design of an electronic circuit to convert the Electrochemical Depolarized CO_2 Concentrator (EDC) generated power into usable electrical energy. This is not the same as power sharing between the EDC and the Water Electrolysis Subsystem (WES) because the EDC power can be used by other electrical devices in the absence

of the WES. Development of actuator failure prediction and detection using audio pattern recognition is recommended. Other advanced designs aiming at further reduction of the instrumentation size and power consumption of Model 200 C/M I is recommended. An example of the advanced designs is multiplexing and transmitting sensor data from the process assembly to the C/M I using a multiplexed transmitter. Evaluation and selection of available ruggedized electronics and space-borne packaging techniques are also strongly recommended.

REFERENCES

1. Ingelfinger, A. L. and Secord, T. C., "Life Support for Large Space Stations," presented at AIAA Fifth Annual Meeting, Philadelphia, PA; October, 1968.

2. Quattrone, P. D.; Babinsky, A. D. and Wynveen, R. A., "Carbon Dioxide Control and Oxygen Generation," presented at the ASME Conference, Paper No. 70-AV/SpT-8, Los Angeles, CA; June, 1970.

3. Wynveen, R. A. and Quattrone, P. D., "Electrochemical Carbon Dioxide Concentrator System," Final Report, Contract NAS2-6118, NASA CR-114426, Life Systems, Inc., Cleveland, OH; March, 1972.

4. Wynveen, R. A.; Schubert, F. H. and Powell, J. D., "One-Man, Self-Contained CO_2 Concentrating System," Final Report, Contract NAS2-6118, NASA CR-114426, Life Systems, Inc., Cleveland, OH; March, 1972.

5. Willis, N. C., Jr. and Neel, J. M., "Space Prototype Environmental Thermal Control and Life Support System - A Current View," presented at the Second Annual Intersociety Conference on Environmental Systems, ASME Paper No. 72-ENAv-35, San Francisco, CA; August, 1972.

6. Marshall, R. D.; Wynveen, R. A. and Carlson, J. N., "Evaluation of an Electrochemical N_2/H_2 Gas Separator," Final Report, Contract NAS2-7057, NASA CR-114580, Life Systems, Inc., Cleveland, OH; March, 1973.

7. Wynveen, R. A. and Schubert, F. H., "A Regenerable Fuel Cell System for Modular Space Station Integrated Electrical Power," Paper No. 739055, Eighth Intersociety Energy Conversion Engineering Conference Proceedings, Philadelphia, PA; August, 1973.

8. Samonski, F. H. and Neel, J. M., "Atmosphere Revitalization for Manned Spacecraft - An Assessment of Technology Readiness," presented at the 24th IAF Congress, Baku, USSR; October, 1973.

9. Quattrone, P. D., "Spacecraft Oxygen Recovery System," Astronautica Acta, Volume 18, No. 4, pp. 261-272; October, 1973.

10. Jensen, F. C. and Schubert, F. H., "Hydrogen Generation Through Static Feed Water Electrolysis," Hydrogen Economy Miami Energy Conference, Miami, FL; March, 1974.

11. Schubert, F. H. and Kostell, G. D., "Six-Man, Self-Contained Carbon Dioxide Concentrator Subsystem for Space Station Prototype (SSP) Application," Final Report, Contract NAS2-6478, NASA CR-114742, Life Systems, Inc., Cleveland, OH; March, 1974.

12. Popma, D. C., "Environmental Control System Development for Manned Spacecraft--1960-1973," presented at the Fourth Annual Intersociety Conference on Environmental Systems, ASME Paper No. 74-ENAs-50, Seattle, WA; July-August, 1974.

13. Wynveen, R. A.; Schubert, F. H.; Kostell, G. D.; Hallick, T. M. and Quattrone, P. D., "Electrochemical Concentration of Carbon Dioxide Aboard Spacecraft," Paper No. 74-109, International Astronautical Federation Congress, Amsterdam, The Netherlands; September - October, 1974.

14. Schubert, F. H. and Quattrone, P. D., "Development of a Six-Man, Self-Contained Carbon Dioxide Collection Subsystem for Spacecraft Application," presented at the Fourth Annual Intersociety Conference on Environmental Systems, ASME Paper No. 74-ENAs-16, Seattle, WA; July-August, 1974.

15. Marshall, R. D., "Spacecraft Nitrogen Resupply Technology: Nitrogen/Hydrogen Separation," Paper 74-ENAs-9, presented at the Intersociety Conference on Environmental Systems, Seattle, WA; July-August, 1974.

16. Schneider, J. J.; Schubert, F. H.; Hallick, T. M. and Woods, R. R., "Electrochemical Carbon Dioxide Concentrator Advanced Technology Tasks," Final Report, Contract NAS2-6478, CR-137732, Life Systems, Inc., Cleveland, OH; October, 1975.

17. Schubert, F. H.; Marshall, R. D.; Hallick, T. M. and Woods, R. R., "One-Man Electrochemical Air Revitalization System Evaluation," Final Report, Contract NAS9-14658, ER-284-4, Life Systems, Inc., Cleveland, OH; July, 1976.

18. Schubert, F. H.; Woods, R. R.; Hallick, T. M. and Heppner, D. B., "Technology Advancement of the Electrochemical Carbon Dioxide Concentrating Process," Annual Report, ER-258-10, Life Systems, Inc., Cleveland, OH; May, 1977.

19. Heppner, D. B.; Hallick, T. M. and Schubert, F. H., "Integrated Operation of a Bosch-Based Oxygen Recovery System," Final Report, Contract NAS8-30891, ER-237-11, Life Systems, Inc., Cleveland, OH; September, 1977.

20. Marshall, R. D.; Lee, M. K. and Schubert, F. H., "Evaluation of a Spacecraft Nitrogen Generator," Final Report, Contract NAS2-8732, CR-152097, Life Systems, Inc., Cleveland, OH; April, 1978.

21. Schubert, F. H.; Lee, M. K.; Davenport, R. J. and Quattrone, P. D., "Water Electrolysis System: H_2 and O_2 Generation," Paper No. 78-ENAs-3, Intersociety Conference on Environmental Systems, San Francisco, CA; July, 1978.

22. Schubert, F. H.; Heppner, D. B.; Hallick, T. M. and Woods, R. R., "Technology Advancement of the Electrochemical CO_2 Concentrating Process," Final Report, Contract NAS2-8666, CR-152250, Life Systems, Inc., Cleveland, OH; May, 1979.

23. Ellis, G. S.; Wynveen, R. A. and Schubert, F. H., "Preprototype Vapor Compression Distillation Subsystem," Final Report, Contract NAS9-15267, ER-312-4, Life Systems, Inc., Cleveland, OH; August, 1979.

24. Powell, J. D. and Kostell, G. D., "Life Support Subsystem Monitoring Instrumentation," presented at the Fourth Annual Intersociety Conference on Environmental Systems, ASME Paper No. 74-ENAs-25, Seattle, WA; July-August, 1974.

25. Yang, P. Y.; Schubert, F. H.; Gyorki, J. R. and Wynveen, R. A., "Advanced Instrumentation Concepts for Environmental Control Subsystems," Final Report, Contract NAS2-9251, ER-309-6, NASA CR-152100, Life Systems, Inc., Cleveland, OH; June, 1978.

26. Yang, P. Y; Gyorki, J. R. and Wynveen, R. A., "Instrumentation for Controlling and Monitoring Environmental Control and Life Support Systems," paper presented at the Eighth Annual Intersociety Conference on Environmental Systems, ASME Paper No. 78-ENAs-40, San Diego, CA; July, 1978.

27. See, G. G.; Yang, P. Y. and Wynveen, R. A., "Development of Instrumentation for a Water Processing Element," Proceedings of the Third National Conference on Complete Water Reuse, Cincinnati, OH; June, 1976.

28. Yang, P. Y.; See, G. G. and Yeh, J. Y., "A Minicomputer-Based Control and Monitor System for Water and Wastewater Treatment," Proceedings of International Computer Symposium, 1977, Volume 2, pp. 856-859, Taipei, Republic of China; December, 1977.

29. Soisson, H. E., "Instrumentation In Industry," John Wiley & Sons, New York, NY; 1975.

30. Avizienis, A., "Fault-Tolerant Systems," IEE Transactions on Computers, Volume C-25, No. 12, pp. 1304-1312, December, 1976.

31. Bailey, S. J., "Man-Process Interface is a Vital Display Function," Control Engineering, Volume 25, No. 3, pp 46-50, March, 1978.

32. "Goals and Guidelines for Control Engineers: 1980," Control Engineering, pp. 58-72, October, 1974.

33. Rivas, J. R. and Rudd, D. F., "Synthesis of Failure-Safe Operations," AIChE Journal, Volume 20, No. 2., pp. 320-325, March, 1974.

34. Sheskin, T. J., "Fault Isolation for Modular Electronic Equipment," Proceedings of Annual Reliability and Maintainability Symposium, pp. 271-274, 1979.

35. "The Configurations of Process Control: 1979," Control Engineering, pp. 43-57, March, 1979.

36. Anderson, P. G., "Redundancy Techniques for Software Quality," Proceedings of Annual Reliability and Maintainability Symposium, pp. 86-93, 1979.

37. Ballard, D. R., "Design Fail-Safe Microprocessor Systems," Electronics, pp. 139-143, January, 1979.

38. Howley, P. P., Jr. "Software Quality Assurance for Reliable Software," Proceedings of Annual Reliability and Maintainability Symposium, pp. 73-78, 1978.

39. Koren, I. and Kohavi, Z., Sequential Fault Diagnosis in Combinational Networks," IEEE Transactions on Computers, Volume C-26, No. 4, pp. 334-342, April, 1977.

40. Littlewood, B., "How to Measure Software Reliability and How Not To," IEEE Transactions on Reliability, Volume R-28, No. 2, pp. 103-110, June, 1979.

41. McPherson, J. A. and Kime, C. R., "A Two-Level Diagnostic Model for Digital Systems," IEEE Transactions on Computers, Vol. C-27, No. 1, pp.16-27, January, 1979.

42. Powers, G. J.; Cummings, D. L. and Rathore, R. N. S., "Synthesis of Fault Tolerant Reaction Paths," AIChE Journal, Volume 21, No. 1, pp. 90-95, January, 1975.

43. Powers, G. J. and Tompkins, F. C., Jr., "Fault Tree Synthesis for Chemical Processes," AIChE Journal, Volume 20, No. 2, pp. 376-387, March, 1974.

44. Teschler, L., "Computers That Won't Fail," Machine Design, pp. 91-97, May, 1978.

45. Ireson, W. G. (Ed.), "Reliability Handbook," McGraw-Hill, New York, NY; 1966.

46. Thayer, R. H. and Hinton, E. S., "Software Reliability - A Method That Works, Proceedings of National Computer Conference, Vol. 44, 1975.

47. Weitzman, C., Minicomputer Systems, Prentice-Hall, Englewood Cliffs, NJ; 1974.

48. Ingels, F. M., Information and Coding Theory, Intext, Scranton, PA; 1971.

49. Duda, R. O. and Hart, P. E., Pattern Classification and Scene Analysis, Wiley, New York, NY; 1972.

50. Hogg, R. V. and Craig, A. T., Introduction to Mathematical Statistics, McMillan, 1965.

51. Fu, K. S., "Pattern Recognition," Computer, IEEE, May, 1976.

52. Lindenlaub, J. C., Guide to Multispectral Data Analysis Using LARSYS, Laboratory for Applications of Remote Sensing (LARS), Purdue University, West Lafayette, IN.

53. You, K. C. and Fu, K. S., "An Approach to the Design of a Linear Binary Tree Classifier," Symposium of Machine Processing of Remotely Sensed Data, Purdue University, West Lafayette, IN; June, 1976.

54. Fu, K. S., Sequential Methods in Pattern Recognition and Machine Learning, Academic Press, New York; 1968.

55. Andrews, H. C., Introduction to Mathematical Techniques in Pattern Recognition, Wiley, New York, NY; 1972.

56. Heffman, G., "Focus on FFT Spectrum Analyzers: Know What Makes Them Measuring Computers," Electronic Design, Volume 26, No. 24, pp. 170-175, November, 1978.

57. Shumar, J. W.; Lantz, J. B. and Schubert, F. H., "The Development of an Electrochemical Technique for In Situ Calibration of Combustible Gas Detectors," Final Report, Contract NAS9-14658, ER-284-13, Life Systems, Inc., Cleveland, OH; July, 1976.

58. Fu, K. S. (ed.), Syntactic Pattern Recognition Applications, Springer-Verlag, 1977.

59. Lewis, P. M. II; Rosenkrantz, D. J. and Stearns, R. E., Compiler Design Theory, Addison Wesley, 1976.

60. You, K. C. and Fu, K. S., "A Syntactic Approach to Shape Recognition Using Attributed Grammer," IEEE Trans. on Systems, Man and Cybernetics, Vol. SMC-9, No. 6, pp. 334-345, June, 1979.

61. "Naked Mini/Alpha LSI Series Computer Handbook," Computer Automation, Inc., No. 91-20400-00A0, Irvine, CA; June, 1974.